National Aeronautics and Space Administration

Goddard Space Flight Center

Code 290, Code 580

IP-in-Space Security Handbook

September 2001

Authored By:

Computer Sciences Corporation

GSFC

Under Contract:

S-38657-G TOR 16

Section 1. Executive Summary

Space Operations Management Office (SOMO), in conjunction with the Operating Missions as Nodes on the Internet (OMNI) project, and in cooperation with Code 291, has tasked Computer Sciences Corporation (CSC) to create a Risk Assessment and Security Handbook with recommendations for securely operating missions as nodes on the Internet. CSC is working with the OMNI project team, and a consortium of other Goddard Space Flight Center (GSFC) personnel and contractors to develop the necessary security solutions.

This handbook consists of three documents that deal in turn with an operations concept, a risk assessment and the results of testing implementations of the operations concepts using Commercial-Off-the-Shelf (COTS) hardware and software. In this handbook, it is assumed that the spacecraft or an instrument on the spacecraft is either National Aeronautics and Space Administration (NASA) owned, or so closely affiliated with NASA that NASA would have a stake in its success or failure.

1.1 Background

In this document, "IP" refers to the Internet Protocol suite and its applications and implementations. "IP-in-Space" refers to the use of IP or the Internet to communicate with a spacecraft.

Traditionally, IP is not used on spacecraft or on the wireless link between a ground station and the spacecraft. However, IP networks are used to transport and route commands and data in some secure NASA ground networks. Prior to command uplink, the communication is translated from IP into other protocols or encoded for transmission to the spacecraft. This paradigm uses expensive proprietary equipment and protocols, and limits access to the spacecraft.

A new paradigm opens up spacecraft to the Internet. Today, missions see benefit in extending IP networks to the spacecraft for two reasons. First, COTS IP products used in spacecraft systems may facilitate cheaper spacecraft and faster development time. Also, by facilitating IP connectivity and data routing from the spacecraft to the Internet, Principle Investigators (PI) or Flight Operations Teams (FOT) could control their experiments or the spacecraft from any IP host. IP would carry communications from the ground end-point, whether an FOT or PI, to the spacecraft, and from the spacecraft back to data archive centers, or directly to end users, possibly viewing spacecraft data with a web client. This may improve accessibility to the spacecraft and its data, and lower ground support costs.

With the benefits of using IP and the Internet to communicate with spacecraft come additional risk due to the open nature of the networks involved and the widespread knowledge of IP and IP implementation and application flaws. The risk assessment section of this document will help each mission that chooses to use IP or the Internet to evaluate that additional risk.

1.2 Special Management Attention

Because IP-in-Space efforts will communicate command and control information, any IP-in-Space mission system is considered a Mission Critical System.[1] As such, all IP-in-Space missions require special management attention, and extra care in developing a secure system.

1.3 Risk Assessment

This document includes an analysis of the additional risk missions incur when choosing to communicate with spacecraft and instruments via the Internet Protocol (IP) and open networks, such as the Internet, rather than traditional proprietary protocols and closed networks. This analysis is a high-level risk assessment that is intended to inform mission planners about:

- information a risk assessment should contain,
- types of issues they need to consider while conducting their own risk assessments, and
- generic risks that may be increased with the use of IP and/or the Internet.

The IP-in-Space mission assets requiring protection can be categorized into types of information and the types of devices that contain, transmit, or generate that information. The material value of the devices and the value of the information should be considered when assessing risk.

Information assets are defined as Mission Uplink, Mission Downlink, and Ordinary Downlink. These categories correspond to the information categories defined in NPG 2810.1. Mission uplink and downlink correspond to the 2810.1 Mission category. Ordinary Downlink corresponds to the 2810.1 categories Business and Restricted Technologies, Scientific and Engineering Research, Administrative, and Public.

The threats and vulnerabilities due to IP increase because of the well-known nature of IP and its applications and implementations and because of the increased connectivity possible when using IP. Standard controls independent from the networking protocol should be used in IP-in-Space missions. However, some IP-specific controls should also be used, in layers, as part of end-to-end security architecture. The security architecture should be an integral part of the design of any IP-in-Space mission and should be considered throughout the design process. Including security in the design of a system is the single most important factor affecting the success of a mission’s security.

Current networks may be classified in terms of what information protection they provide by virtue of their current controls. Current networks should be qualified as belonging to one of three defined “control domains:” Closed Mission Network, Open Mission Network, or Open Network. Missions can then use existing networks and understand what protection these networks afford.

1.4 Definitions

The operation concepts and their associated scenarios are described using the information categories, network definitions and security attributes defined in the following sections.

[1] NPG 2810.1 Section 4.2.8, bullet c.

1.4.1 Information Category

Missions contain a variety of information types, and these different types of information travel over different paths in the networks. The path, its security characteristics, and the information type all are considered when determining risk. The NPG 2810.1 specifies five information types into which all of the mission information should be categorized. These are:

- Mission (MSN)
- Business and Restricted Technology (BRT)
- Scientific and Engineering Research (SER)
- Administrative (ADM)
- Public (PUB)

Any uplink, or communication destined for the spacecraft, including commands, is categorized as Mission (MSN) Information. Some downlink information, or information originating at the spacecraft, will also fall into the MSN category. Any other downlink may be BRT, SER, ADM, or PUB information. From this point forward, this document will use the following terms to categorize information:

- ***Mission Uplink***: This includes all uplink, or communication destined for the spacecraft, including command data. This information is considered MSN.
- ***Mission Downlink***: This includes any downlink, or communication, originating at the spacecraft that can be considered MSN. Spacecraft and Instrument Engineering telemetry may be classified as mission downlink. Examples are any downlink including IP addresses on-board the spacecraft and memory dumps that may include encryption keys.
- ***Ordinary Downlink***: This includes any downlink, or communication originating at the spacecraft that can be considered BRT, SER, PUB, or ADM information. Ordinary downlink is likely to include science telemetry, but may also include some instrument or spacecraft engineering telemetry.

1.4.2 Networks

The operations concept scenarios presented in this document describe a communication path using the following three network types. The first network type is a Closed Mission Network path (CMN) that has been certified by NASA as meeting NPG 2810.1 requirements for mission (MSN) information, connected to only an open mission network, and connected to that open mission network via a stateful firewall. The second network type is an Open Mission Network path (OMN) that has been certified by NASA as meeting NPG 2810.1 requirements for MSN information; but is connected to other networks that may not fulfill the NPG 2810.1 requirements for MSN information. The third network type is an Open Network path (ON) that has not been certified by NASA as a network that meets NPG 2810.1 MSN requirements, and is accessible by persons unaffiliated with mission operations.

1.4.3 Security Attributes

A security attribute is a type of protection that is afforded to information. This document associates these attributes with each of the information categories from the NPG 2810.1. These associated attributes are then used to select COTS products to implement the paths required by the four scenarios since COTS products are generally categorized by these attributes. The selected products are then assembled to implement a communication path for the operation concept scenario being tested.

Definitions of security attributes used in this document are:

Attribute	**Definition**
Access control	Process of allowing or disallowing access based on many criteria, including but not limited to successful authentication. Access control protects against unauthorized use of a network or networked resource.
Availability	Methods to maintain availability prevent the denial of service through degradation of network services.
Authentication	Establishment of the identity of an entity (either a user or a computer system). Authentication in this document also includes non-repudiation.
Non-repudiation	Non-repudiation is the provision that an entity sending or receiving a communication cannot later deny sending or receiving that communication.
Confidentiality	Methods to maintain confidentiality protect information from being disclosed to unauthorized entities.
Data integrity	Methods to maintain data integrity detect (but do not necessarily prevent) the unauthorized modification or deletion of data.
Traffic flow integrity	Methods to maintain traffic flow integrity prevent the collection of sensitive information about the network through observation of network traffic characteristics. This includes gaining information about the network based on when traffic does or does not flow or based on packet headers being sent to, from, or within the network.

Table 1. Security Attribute Definitions

1.5 Operation Concept Scenarios

Four likely routes for IP communications with a spacecraft were chosen. These are presented as spacecraft operations concept scenarios. Solutions focused on these as the most likely that NASA will need to learn how to protect.

The first scenario is the traditional approach to communicating with a spacecraft. A workstation on a closed mission network, such as the Closed IONet, sends commands to, and receives data from the spacecraft.

The second scenario is communicating over an open mission network through a closed mission network to the spacecraft is similar to communicating over the Open IP Operational Network (IONet) through the Closed IONet.

The third scenario builds on the previous concepts. It addresses all paths except the open networks between the Open Mission Network and the host sending communications.

The final scenario is the vision for the future of IP-in-Space. In this scenario, either the spacecraft, or some portion of the spacecraft, or some information is owned by or affiliated with NASA. However, all of the intermediate networks, including the ground station are open networks. This scenario may include multiple support routes. Missions that require large amounts of data to be downlinked and/or multiple communication contacts per orbit may need to use both NASA and non-NASA ground stations. Mission formulators need to understand that different security policies may apply to their mission depending on the support infrastructure being used. NASA may restrict the use of their assets depending upon the risks associated with the mission requesting support.

1.6 Scenario Testing

COTS products were assembled to determine the feasibility of implementing each operations concept scenario. These products included a Cisco PIX firewall, a Netscreen firewall, and two 3Com encryption capable Network Interface Cards (NICs). A SecurID ACE server and SecurID tokens from another project were also used. BlueCat Linux, an embedded version of Linux, was used to simulate a spacecraft. And, many freeware packages were used to implement the operations concept scenarios.

Testing efforts were conducted in several different labs. Whenever possible, testing observations were focused on the technologies. This testing was not meant to differentiate between vendors' implementations of the technologies or to endorse any one vendor over any other. The purpose was to determine if a solution with a mix of technologies and tools could be built to allow the necessary spacecraft communication and provide security.

Each scenario was configured so that traffic could flow from the simulated control center host to the simulated spacecraft, through the appropriate network paths, and with the appropriate security technologies in place. These scenarios were set up as a proof-of-concept to show that the technologies proposed could be configured to work together as indicated in the scenarios. To the greatest extent possible, certain common elements were measured in each effort. These elements are:

- CPU and/or Memory Utilization
- Latency or Delay
- Bandwidth utilization and/or overhead
- Complexity or Ease of Use and Interoperability (subjective measure)

    - Platforms the tools run on
    - How well the tool works with products from other vendors and with other technologies
- Features of Security
    - Protection the technology provides and to what extent
    - Difficulty or Ease of subverting the security technology
    - NIST/NSA NIAP certified or other standards compliant?
- Stability of the technology

The rest of this document develops the proposed solutions that were tested, and presents the results of the tests, along with recommendations for future efforts that will help find the more secure solutions for using IP and the Internet to communicate with spacecraft.

1.7 Recommendations

The following are recommendations derived from the analyses and testing that are the basis of this document. The recommendations are in two categories: risk assessment and technology selection.

1.7.1 Risk Assessment

On Open Networks (such as the Internet), the risk of IP-in-Space is greatest and the possible mitigation may be difficult and costly. Therefore, we recommend that Mission Uplink, including spacecraft command, should not be transmitted over Open Networks at this time. On Open Mission Networks, risk to Mission Uplink data may be more effectively mitigated. Therefore, under special conditions, spacecraft commands may be transmitted over Open Mission Networks. The wireless link from the ground station to the spacecraft also warrants additional protection; however the risk is not so large that commanding spacecraft over this link using IP should be prohibited. It may be acceptable under certain conditions. Closed Mission Networks are acceptable for transmitting IP-in-Space communications.

1.7.2 Technology Recommendations

IP standards-based COTS security products can be used together to create communication systems with added protection against risk. We found that creating an integrated solution with many products can be difficult and complex. To simplify the security solution and improve system efficiency, the security should be designed into the system rather than applied after the fact.

Encryption does affect CPU resources and add significant overhead. The amount of processor utilization and packet overhead varies depending on the encryption algorithm, the IPSec mode, and the size of the unencrypted packets. Each mission will have to design their systems with these security mechanisms in mind in order to develop the most efficient set of packet size, encryption algorithm, and IPSec mode for their applications. Hardware encryption on-board the spacecraft should be explored.

Bandwidth is reduced when encryption is employed. Missions will have to consider their data rate requirements and plan their processor and memory resources accordingly, so that they have enough resources to encrypt and decrypt data at the rate they require.

Our proposed solutions had difficulty meeting or could not meet the recommended protection requirements for communication paths outside redundant mission networks. Outside the mission network domains, availability is unreliable and sufficient protection of the Primary Investigator (PI) or Flight Operations Team (FOT) host computer is questionable. NASA will have to weigh the risks involved to determine if a host on an open network should have commanding access to the spacecraft or spacecraft instrument, and if so, to determine how to protect that host.

National Aeronautics and Space Administration

Goddard Space Flight Center

Code 290, Code 580

IP-in-Space Security Operations Concepts

March 2001

Authored By:

Computer Sciences Corporation

GSFC

Under Contract:

S-38657-G TOR 16

Contents

List of Figures

List of Tables

Section 1. Introduction

Space Operations Management Office (SOMO), in conjunction with the Operating Missions as Nodes on the Internet (OMNI) project, and in cooperation with Code 291, has tasked Computer Sciences Corporation (CSC) to create a Risk Assessment and Security Handbook with recommendations for securely operating missions as nodes on the Internet. CSC is working with the OMNI project team, and a consortium of other Goddard Space Flight Center (GSFC) personnel and contractors to develop the necessary security solutions.

This document is intended as the first step towards creating the IP-in-Space Risk Assessment and IP-in-Space Security Handbook. In this document, the possible paths used in Internet Protocol (IP) communication with a spacecraft will be characterized. Likely scenarios, or operations concepts, for using IP to communicate with a spacecraft over subsets of those possible paths will be identified. While the risk assessment and handbook are meant to be guides for all missions using IP, they cannot contain security solutions for all possible missions. Therefore these operations concepts are meant to serve as illustrative examples, providing a focus for the project effort and the resulting documentation.

In this document, it is assumed that the spacecraft or an instrument on the spacecraft is either National Aeronautics and Space Administration (NASA) owned, or so closely affiliated with NASA that NASA would have a stake in its success or failure.

Section 2. Communications Paths

2.1 Path Characteristics

As a host communicates a spacecraft, the communication travels a route over many different paths. For our purposes, a “path” will be a section of network with similar characteristics. A “route” will be the combination of paths traversed between the host and the spacecraft. A path may be characterized by: its level of NASA affiliation, by the type of its physical link (copper, fiber, wireless), and by the openness of the network in which it resides. This section describes the different path characterizations, and determines candidate paths that should be more thoroughly analyzed to determine the security implications.

2.1.1 NASA Affiliation/ownership

How much responsibility NASA has for the maintenance of a path may determine its level of security. If NASA is involved in maintaining a path, then NASA security requirements may be enforced on that path, enabling secure IP spacecraft communications without much additional effort on the part of the mission/project. If NASA is not involved at all in a path, then NASA may not have control over security in that path. NASA may have to find a different solution to protect traffic as it traverses that path. This solution may include some form of certification that the network does meet the NASA Policy and Guidelines (NPG) 2810.1[1] requirements for Mission MSN information, even if it is not obligated to do so. The solution could be some kind of technology layered on top of the path that helps secure NASA assets. Therefore, the two main categories of NASA affiliation are divided according to whether or not NASA policy is applicable. This distinction is not cut-and-dry because sometimes NASA policy can not effectively be enforced in all teaming arrangements, especially with international partners due to international laws and regulations and export agreements and restrictions.

- NASA Policy Applicable
  - Wholly owned (with the exception of dedicated leased Wide Area Network links) and wholly operated by NASA or its contractors.
  - Mostly NASA operated, but with some teaming arrangements. Partners accept and abide by NASA security policy wherever possible, subject to international law complications and approved waivers.
- NASA Policy Not Applicable
  - NASA does not operate the network or mission, but a perception exists that the network or mission is affiliated with NASA

[1] http://nodis.hq.nasa.gov/Library/Directives/NASA-WIDE/Procedures/Legal_Policies/N_PG_2810_1/contents.html

    - Not NASA affiliated. (e.g. a commercial groundstation or mission)

2.1.2 Link characterization

Link characterizations are:

- Wireless link:
    - satellite relay link
    - Personal Digital Assistant (PDA) wireless link to a LAN
    - “Internet enabled” mobile-phone
- Terrestrial Link:
    - local area copper
    - local area fiber
    - Wide-area fiber

In many cases, the link type is irrelevant when considering security. All of these links may be tapped. However, it is easier to intercept wireless signals, whether coming from a hand-held PDA, or from a satellite dish. Wireless technologies may be tapped from the air. Terrestrial technologies must be tapped at the physical point of some cabling. Therefore, wireless and terrestrial will be the two categories of link-type considered.

2.1.3 Openness

The degree to which a network path is open, or exposed to the public, is a characteristic which impacts security considerations. The following categories of “openness” will be considered.

- Air-Gap: An air-gap path exists on a collection of equipment interconnected with a network that connects to nothing else except the Radio Frequency (RF) link to the spacecraft. The RF link itself is not considered an air-gap path. Wireless paths cannot be air-gap paths, as the wireless element allows connectivity over the air, voiding the premise of an air gap.

- Closed Mission Network: A closed mission network path is one which has been certified by NASA as fulfilling NPG 2810.1 requirements for mission (MSN) information, connected to only an open mission network, and connected to that open mission network via a stateful firewall. Wireless devices cannot exist in closed mission networks, because the wireless access may provide an entry point into the network that could bypass the firewall. The firewall may be a system that may include external and internal routers, firewall bastions, authentication mechanisms, and other protection devices which control the input and output of information between the open mission network and the closed mission network. The firewall system must be capable of creating and maintaining a log or audit trail of all packets that flow between the protected closed mission network and the open mission network. The firewall system must also maintain state, such that it can recognize a packet as a legitimate part of a communication. It must selectively allow access based on International Standards Organization (ISO) Open Systems Interconnection (OSI) layer 3 and 4 information, and only allow access as a response to an outgoing communication from the closed mission network.

- Open Mission Network: An open mission network path is one that has been certified by NASA as fulfilling NPG 2810.1 requirements for MSN information. It connects to other networks that may not fulfill the NPG 2810.1 requirements for MSN information. It is protected from these other networks by a router with port and IP address filters, but not by a stateful firewall as described in the description of a closed mission network. It is therefore sheltered from the open networks, but is still more vulnerable than if it were protected by such a strong firewall.

- Open network: An open network path is one which has not been certified by NASA as a network that fulfills NPG 2810.1 MSN requirements, and is accessible by persons unaffiliated with mission operations. A subset of these are modem connections -- dial-in connections, over the public telephone system, into a network.

2.2 Resultant Paths

The possible routes for a communication between a host and the spacecraft can vary greatly. Including the categories and subcategories of the above types of NASA affiliation, link characterization, and degree of openness, there are up to 40 different permutations of possible path types.

From a security perspective, however, many types of paths can be treated similarly, thus narrowing down the requirements that must be levied in order to secure IP communications with spacecraft. For instance, fiber and copper cables can both be tapped. Therefore, terrestrial links may be treated as one category. Such categorization narrows down the path types to 12 possible links, shown in the following table:

Table 2-1: Possible Communication Paths

Path	Path Characteristics		
1	NASA Policy Applicable	Wireless	Open Mission Network
2	NASA Policy Applicable	Wireless	Open Network/Modem
3	NASA Policy Inapplicable*	Wireless	Open Mission Network
4	NASA Policy Inapplicable	Wireless	Open Network
5	NASA Policy Applicable	Terrestrial	Air-Gap
6	NASA Policy Applicable	Terrestrial	Closed Mission Network
7	NASA Policy Applicable	Terrestrial	Open Mission Network

Path	Path Characteristics		
8	NASA Policy Applicable	Terrestrial	Open Network/Modem
9	NASA Policy Inapplicable	Terrestrial	Air-Gap
10	NASA Policy Inapplicable*	Terrestrial	Closed Mission Network
11	NASA Policy Inapplicable*	Terrestrial	Open Mission Network
12	NASA Policy Inapplicable	Terrestrial	Open Network/Modem

* These paths are only possible if the network has been certified as compliant with the NPG 2810.1 MSN requirements even though the network is not obligated to comply.

Section 3. Operations Concepts

Table 2-1 indicates all the possible data paths along a route between a host and the spacecraft. This section highlights 4 routes through the most likely sampling of those data paths. These are presented as spacecraft operations concepts. Solutions and requirements will focus on these as the most likely scenarios that NASA will need to learn how to protect. A study will follow to produce security solutions for these different scenarios, in the priority order that they are presented.

3.1 Concept 1: Communication Over Closed Mission Networks

Traditional approach, with a workstation on a closed mission network, such as the Closed IONet.

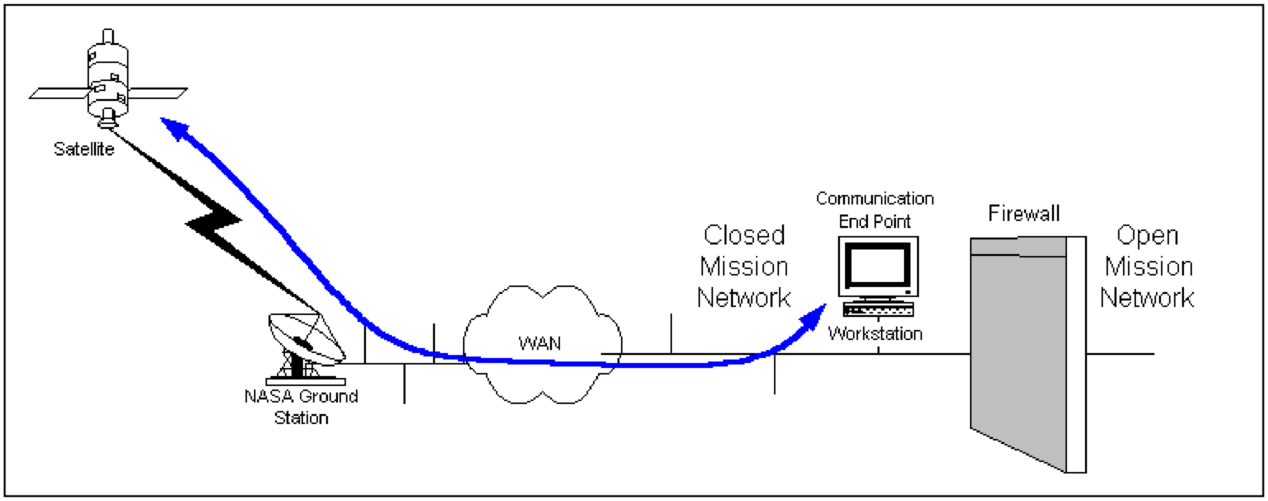


Figure 3-1: Communication over Closed Mission Networks

Many missions currently use IP to carry communications within the bounds of closed mission networks, as shown in this concept. However, the last path of this communication route to the spacecraft, the RF link between the ground and the spacecraft, currently does not use IP. To use IP in this path may require additional security measures. Because this RF link is common to all IP-in-Space scenarios, this first scenario has the highest priority in the study. Solutions for this scenario may also apply to communications over air-gap network paths.

3.2 Concept 2: Communication Over Open Mission Networks

Communicating over an open mission network, through a closed mission network, to the spacecraft is similar to communicating over the Open IP Operational Network (IONet) through the Closed IONet. The network segments indicated are Ethernet, but they could be any type.

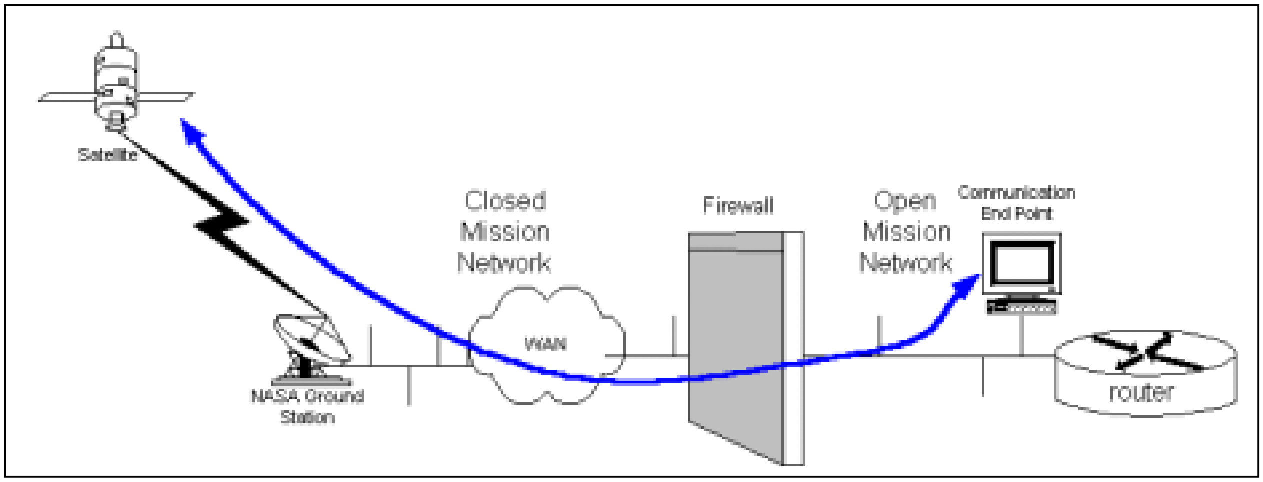


Figure 3-2: Communication over Open Mission Networks

Today, commanding is not permitted on the Open IONet. Commanding over an open mission network will require additional security measures. Because it is likely that the first step towards operating spacecraft as nodes on the internet will include commanding over a sheltered, if not isolated LAN, this scenario is considered as the second priority in the study.

3.3 Concept 3: Communication Over an Open Network

This scenario builds further on the above concepts. Solutions for the above concepts will address all paths of this concept except those open networks between the Open Mission Network and the host sending communications.

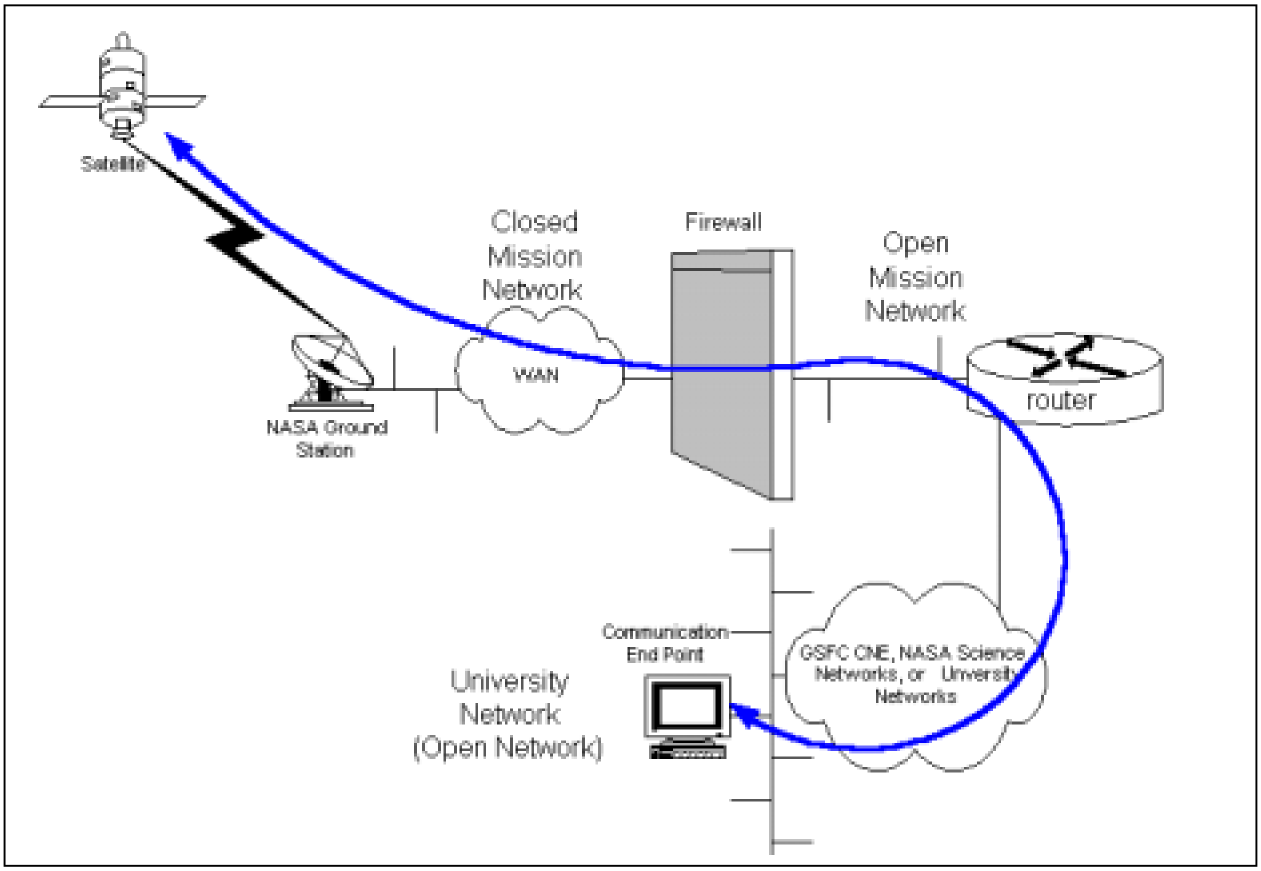


Figure 3-3: Communication over Open Networks

The open network medium indicated is Ethernet, but could be a different link type, or a mixture of link types. The important fact is that persons who are not affiliated with the mission could intercept traffic flowing on the open networks. Because NASA often coordinates with universities and other scientific partners, this is a likely scenario for IP-in-Space missions. It is considered the third priority in the study.

3.4 Concept 4: Communication Over the Internet

This last concept is the vision for the future of IP-in-Space. In this concept, either the spacecraft, or some portion of the spacecraft, or some information is owned by or affiliated with NASA. However, all of the intermediate networks, including the ground station are open networks.

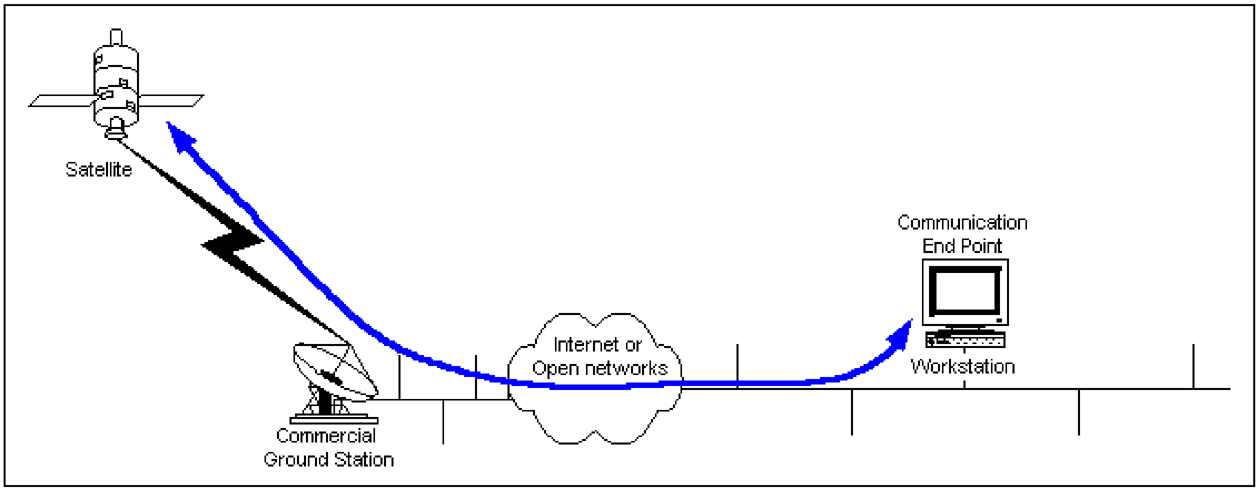


Figure 3-4: Communication over the Internet

This is the riskiest scenario, and will require a multifaceted, layered security solution; it therefore has the lowest priority in the study and may not be fully addressed. This scenario may include multiple support routes. For missions that require large amounts of data to be downlinked and/or multiple communication contacts per orbit may need to use both NASA and non-NASA ground stations. Mission formulators need to understand that different security policies may apply to their mission depending on the support infrastructure being used. NASA may restrict the use of their assets depending upon the risks associated with the mission requesting support.

National Aeronautics and Space Administration

Goddard Space Flight Center

Code 290, Code 580

IP-in-Space Risk Assessment & Information Protection Recommendations

June 2001

Authored By:

Computer Sciences Corporation

Under Contract:

S-38657-G TOR 16

With Contributions From:

GSFC

Lockheed Martin Corporation

Executive Summary

This Risk Assessment document highlights the additional risk missions incur when choosing to communicate with spacecraft and instruments via the Internet Protocol (IP) and open networks, such as the Internet, rather than traditional proprietary protocols and closed networks. Each mission is responsible for conducting a risk assessment of its own system(s). This document is a high-level risk assessment, which is intended to inform mission planners about:

- the information a risk assessment should contain,
- the types of issues they need to consider while conducting their own risk assessments, and
- the generic risks that may be increased with the use of IP and/or the Internet.

In this document, "IP" refers to the Internet Protocol suite and its applications and implementations. "IP-in-Space" refers to the use of IP or the Internet to communicate with a spacecraft.

The IP-in-Space mission assets requiring protection can be categorized into types of information and the types of devices that contain, transmit, or generate that information. The material value of the devices and the value of the information should be considered when assessing risk.

Information assets are defined as Mission Uplink, Mission Downlink, and Ordinary Downlink. These categories correspond to the information categories defined in NPG 2810.1. Mission uplink and downlink correspond to the 2810.1 Mission category. Ordinary Downlink corresponds to the 2810.1 categories Business and Restricted Technologies, Scientific and Engineering Research, Administrative, and Public.

The threats and vulnerabilities due to IP increase because of the well-known nature of IP and its applications and implementations and because of the increased connectivity possible when using IP. Standard controls independent from the networking protocol should be used in IP-in-Space missions. However, some IP-specific controls should also be used, in layers, as part of an end-to-end security architecture. The security architecture should be an integral part of the design of any IP-in-Space mission and should be considered throughout the design process. Including security in the design of a system is the single most important factor affecting the success of a mission's security.

Current networks may be classified in terms of what information protection they provide by virtue of their current controls. Current networks should be qualified as belonging to one of three defined "control domains:" Closed Mission Network, Open Mission Network, or Open Network[1]. Missions can then use existing networks and understand what protection these networks afford.

[1] These are defined in detail in section 5.

On Open Networks (such as the Internet), the risk is greatest and the possible mitigation may be difficult and costly. Therefore, we recommend that Mission Uplink, including spacecraft command, should not be transmitted over Open Networks at this time. On Open Mission Networks, risk to Mission Uplink data may be more effectively mitigated. Therefore, under special conditions, spacecraft commands may be transmitted over Open Mission Networks. The wireless link from the ground station to the spacecraft also warrants additional protection; however the risk is not so large that commanding spacecraft over this link using IP should be prohibited. It may be acceptable under certain conditions. Closed Mission Networks are acceptable for transmitting IP-in-Space communications.

An IP-in-Space Security Handbook will be produced to demonstrate some possible solutions for successfully mitigating risk in some IP-in-Space scenarios. This handbook may be used as a starting point for IP-in-Space missions designing their security architectures. It should be a living document, with input from all IP-in-Space missions, so that NASA can leverage all its resources to facilitate a stronger security posture for all IP-in-Space missions, and eventually secure spacecraft command communications over the Internet.

Contents

Section 1. Introduction

1.1 Background

Traditionally, IP is not used on spacecraft or on the wireless link between a ground station and the spacecraft. IP networks are used to transport and route commands and data in some secure NASA ground networks. However, prior to command uplink, the communication is translated from IP into other protocols or encoding for transmission to the spacecraft. This paradigm uses expensive proprietary equipment and protocols, and limits access to the spacecraft.

A new paradigm opens up spacecraft to the Internet. Today, missions see benefit in extending IP networks up to the spacecraft for two reasons. First, Commercial Off-the-Shelf (COTS) IP products used in spacecraft systems may facilitate cheaper spacecraft and faster development time. Also, by facilitating IP connectivity and data routing from the spacecraft to the Internet, Principle Investigators (PI) or Flight Operations Teams (FOT) could control their experiments or the spacecraft from any IP host. IP would carry communications from the ground end-point, whether an FOT or PI, to the spacecraft, and from the spacecraft back to data archive centers, or directly to end users, possibly viewing spacecraft data with a web client. This may improve accessibility to the spacecraft and its data, and lower ground support costs.

With the benefits of using IP and the Internet to communicate with spacecraft come additional risk due to the open nature of the networks involved and the widespread knowledge of IP and IP implementation and application flaws. This risk assessment will help each mission that chooses to use IP or the Internet to evaluate that additional risk.

1.2 Scope

It is recommended that each mission perform a risk assessment of their mission-specific development and operational environments. This risk assessment is intended to be a generic IP-in-Space Risk Assessment baseline, which can be used by missions as a starting point for their mission-specific risk analyses. Because the scope is broad, the major threats, vulnerabilities, and resultant risks included in this document are high level, and apply to any mission using IP.

1.3 Methodology

The methodology used in this risk assessment, and recommended for individual missions' risk assessments, closely follows the methodology presented in the NPG 2810.1 document. Threats and vulnerabilities together define risk. "If a system or application is vulnerable to a threat, it is considered a risk."[2] When threats and vulnerabilities are considered, along with the value of the resources at risk, and the security controls used to counter the threats and vulnerabilities, the level of risk may be determined. Generic threats, vulnerabilities, controls, and the resultant risks involved in using IP to communicate with spacecraft are described in this assessment.

[2] NPG 2810.1 Chapter 4.

1.4 Responsible Official and Staff

The responsible official and staff will vary from mission to mission. Each mission must identify these critical personnel for all portions of the mission system. If the risk assessment covers the entire mission, this section should minimally include the appropriate Computer Security Officer(s) (CSO), the Mission Director, the Project Manager, and any relevant system and network administrators.

1.5 Special Management Attention

Because IP-in-Space efforts will communicate command and control information, any IP-in-Space mission system is considered a Mission Critical System.[3] As such, all IP-in-Space missions require special management attention, and extra care in developing a secure system.

1.6 Information Category

Missions contain a variety of information types, and these different types of information travel over different paths in the networks. The path, its security characteristics, and the information type all are considered when determining risk. The NPG 2810.1 specifies five information types into which all of the mission information should be categorized. These are:

- Mission (MSN)
- Business and Restricted Technology (BRT)
- Scientific and Engineering Research (SER)
- Administrative (ADM)
- Public (PUB)

Any uplink, or communication destined for the spacecraft, including commands, is categorized as Mission (MSN) Information. Some downlink information, or information originating at the spacecraft, will also fall into the MSN category. Any other downlink may be BRT, SER, ADM, or PUB information. From this point forward, this documentation will use the following terms to categorize information:

- ***Mission Uplink***: This includes all uplink, or communication destined for the spacecraft, including command data. This information is considered MSN.
- ***Mission Downlink***: This includes any downlink, or communication, originating at the spacecraft that can be considered MSN. Spacecraft and Instrument Engineering telemetry may be classified as mission downlink. Examples are any downlink including IP addresses on-board the spacecraft and memory dumps that may include encryption keys.
- ***Ordinary Downlink***: This includes any downlink, or communication originating at the spacecraft which can be considered BRT, SER, PUB, or ADM information. Ordinary

[3] NPG 2810.1 Section 4.2.8, bullet c.

downlink is likely to include science telemetry, but may also include some instrument or spacecraft engineering telemetry.

The NPG 2810.1 requirements for each type of information (MSN, BRT, SER, ADM, and PUB) are in no way voided by this documentation. This risk assessment will determine risk to assets and information with respect to the NPG 2810.1 categories. Missions will ultimately need to adhere to both the NPG 2810.1 policy requirements as well as any other applicable NASA policy requirements. As is normally the case, where more than one policy is applicable, the stricter security policy should be followed, so that all policy requirements are met.

Section 2. Assets at Risk

2.1 Assets

Assets can be material equipment, key personnel, or information. An asset is anything that has value for NASA or NASA stakeholders. The assets of the mission should be identified and valued such that the appropriate amount of risk mitigation can be determined to protect each asset. This step will help missions identify what they have to protect, and the amount of time and money that should be spent to mitigate the risk of losing or damaging the asset.

2.2 Nature and Value of the Assets

Mission budget, level of NASA affiliation, data sensitivity and data value all are factors to consider when estimating the value of an asset. A value can be both quantitative (e.g., monetary replacement value) or qualitative (e.g., NASA's reputation is at stake).

The overall value of an asset depends somewhat on the total science package for the mission's spacecraft. Depending on the mission goals, the value of the assets may be expressed individually or collectively. In the case where a single instrument can be lost and the mission can still produce results, the value of that instrument and the information it produces should be specified individually. In the case where all component assets are needed for a mission to succeed, the assets may be valued together, as the overall value of the mission. A mission with many spacecraft instruments may value any one instrument differently than a mission with a single instrument spacecraft. The more fully loaded spacecraft may assess a value for a single spacecraft instrument that is lower than the cumulative value of the mission.

There is also the situation where a mission associates a high value to a group of instruments on a complex spacecraft. The results of each instrument within the group are needed to derive metadata for an observation. For instance, resource monitoring may require the spectral recordings of several instruments to detect the resource's signature. The value of this observation has to be factored into the value for each instrument contributing to it.

Constellations of spacecraft also must be assessed with respect to the value of a single spacecraft within the constellation and/or the value of the constellation as a whole.

Some assets may also be considered national resources. Federal policy may help missions determine how to assess value of national resources.

Table 2-1 contains generic assets with a process to determine each asset's value based on replacement cost and infrastructure support costs. The table is followed by definitions of each asset.

Table 2-1: Asset Valuation

NASA Asset	Information Type	Value
Science information	Ordinary Downlink	Minimum is the cost of operation of the infrastructure that will process and use the science information.
Engineering Information	Mission or Ordinary Downlink	Minimum is the cost of impact if data (Ordinary Downlink) is lost. This could potentially be the value of the mission if any information (Mission Downlink) lost or accessed by unauthorized parties leads to the destruction of the spacecraft or its instruments.
Spacecraft	contains all types	Cost of replacement through orbit placement (includes launch) plus the cost associated with the science information
Spacecraft instrument	may contain all types	Cost of integration to a spacecraft and a share of the orbit placement plus the cost associated with the science information
Constellation	Contains all types	Cost of replacement through orbit placement (includes launch) plus the cost associated with the science information
Networks	may contain all types	The replacement cost of the network infrastructure, plus any cost of the loss of data due to the loss of the network infrastructure.

Science information is the science data from an on-board instrument together with any required ancillary telemetry data. Ancillary data can include time stamps and complementary housekeeping data. The utility of science data can be degraded if the associated ancillary data is not present. For instance, calibrations to the science instrument may be impossible without spacecraft voltage information.

Engineering Information is the information about the spacecraft itself, such as voltage or temperature measurements, attitude or orbit coordinates, or memory dumps. This information includes command and control information.

Spacecraft include all autonomous orbiting packages and on-board support systems that NASA owns or has some other vested interest in.

A *spacecraft instrument* is a spacecraft component that measures the present value of a quantity (generally science data) under observation. It produces data supplementing the housekeeping data.

Networks comprise the terrestrial equipment such as routers, switches, hubs, cables, radio frequency (RF) dishes, and hosts that interconnect to provide communication with the spacecraft. This includes the Mission Operations Control Center facilities (MOCC) and control center systems, voice systems, and other ground communications system components.

In general, Mission Uplink and Mission Downlink information, and the equipment that contain or generate that information are valued more highly than Ordinary Downlink information. This is due to the fact that loss or corruption of Mission information may lead to the end of a mission, and to the loss of future Ordinary Downlink information. Loss of Ordinary Downlink information is costly, but does not necessarily prevent the continuation of the mission and the collection of future Ordinary Downlink information.

Section 3. Threats

3.1 Categorized Threats

A threat is a circumstance or event that has the potential to cause harm to a computer or network facility or computer/communication system. Threats are generally categorized as either human threats or environmental threats. Human threats can be intentional (e.g., deliberate malicious acts) or unintentional (e.g., errors due to negligence). Environmental threats can be natural or fabricated, (i.e., man- or machine-caused events or mechanical/structural defects).

More details and examples of these threats may be found in Appendix A. Missions should use the threats listed in Appendix A as a starting point to determine what threats may face their own systems. Mission planners should also consult National Security resources[4] to determine the true threat to their mission. This is especially the case for any spacecraft or mission considered a national security resource.

Many threats are not specific to a network communications protocol, and are faced by spacecraft systems today. These threats currently demand consideration when evaluating the risk of a spacecraft system, and will continue to demand consideration when IP is used in future spacecraft systems.

All of the threats relating to physical harm or to attacks by persons with authorization to systems or areas are independent of the network protocol used. Physical harm can be done regardless of the network protocol used. Any attack perpetrated by an insider[5] is also largely unaffected by the protocol used. The threat of terrorism against space systems increases with IP, even though traditionally, terrorist attacks are physical in nature. For instance, terrorism usually involves an attempt to frighten and intimidate, which can most effectively be done by destroying life. This could be done with an IP attack (by gaining control of a manned spacecraft and crashing it or controlling an unmanned spacecraft and de-orbiting it such that it may land in a populated area). Terrorism may also take the form of destroying an unmanned spacecraft without any loss of life, especially if that spacecraft is seen as an important national asset.

3.2 Threats Specific to IP

All of the threats listed in Appendix A exist to some degree, regardless of the protocol used. However, the use of IP increases the means for a person with malicious intent to cause harm. IP systems and attacks on IP systems are prevalent and well known in today's Internet-based culture. The means or methods used to carry out threats against IP systems are large in scope. If an outsider can gain access to a closed network, he or she is better able to do electronic damage if that closed network is using a well-known protocol such as IP. In addition, IP facilitates

[4] agencies or experts dealing in national resource protection.
[5] someone with authorization to access areas or information

connections to a greater number of other Information Technology (IT) resources, increasing the avenues over which an attack may be launched.

Therefore, threats to a mission are increased when using IP and the Internet to communicate with spacecraft. The means to carry out an attack is increased both by the well-known nature of the protocol and common exploits, as well as the increased connectivity to unknown and possibly malicious entities. This causes an increased risk, which warrants additional security measures for IP missions.

Section 4. Vulnerabilities

4.1 Vulnerable Points

A vulnerability is a flaw or weakness in a system that can be exploited to violate security processes or controls. In this case, the system is the spacecraft, the terrestrial hosts it communicates with, and the entire network of communications paths between those hosts and the spacecraft. There are several vulnerable places in this system that an attacker can direct effort in order to achieve a desired effect. Those vulnerable points are:

1. The spacecraft
2. Wireless or Inter-Satellite Link (ISL) communications with the satellite, or with an intermediary relay satellite such as a Tracking and Data Relay Satellite (TDRS)
3. Data paths between ground stations and control centers
4. Ground Station and Control Center computers
5. Communications links between Ground Station or Control Center computers and any remote access hosts
6. Remote access hosts or remote clients

The remote access host connectivity referenced in points 5 and 6 is a new capability which IP-in-Space efforts hope to deliver. Points 2, 3, and 5 are all network links that can be tapped for eavesdropping, insertion, or interference. Points 1, 4, and 6 can all be seen as networked hosts that may be compromised through various vulnerabilities specific to each host.

Categorized examples of vulnerabilities that apply to most systems, and factors that affect their severity, are given in Appendix B. As missions review their own architectures and networked devices for these generic vulnerabilities and for vulnerabilities that arise from their design and implementation, each of the above vulnerable points should be considered. Security is only as strong as its weakest component. Every point of entry should be made strong against attack.

The NPG 2810.1 Appendix A should also be consulted to determine what NASA considers to be the limits of acceptable vulnerability. For example, by comparing the system’s policy on passwords with the NPG 2810.1 policy on passwords, the analyst may judge whether or not NASA would consider the system too vulnerable based on its password management.

4.2 Vulnerabilities Specific to IP

The use of IP increases system vulnerability; particularly if IP is used to provide increased connectivity for the system. The use of IP in future mission scenarios is in part driven by a desire for more access options. The Internet is an IP network, and this IP network now extends into many homes and almost all businesses and universities. Therefore, the use of IP makes it easy to extend a spacecraft communications network to universities and to scientist or FOT homes. When this added connectivity is in place, the vulnerability of a system increases because the connections to so many other Internet hosts expose the system to more threats.

Vulnerabilities also increase with the use of IP due to flaws in IP applications or implementations. While any protocol or application may have flaws that introduce system vulnerabilities, those vulnerabilities introduced into systems by applications and implementations of IP are likely to be better known than other network-protocol implementation vulnerabilities.

The vulnerabilities presented in Appendix B may be used as a guide for missions to consider what typical vulnerabilities may appear in their systems. Tools that will scan systems for well-known vulnerabilities and Intrusion Detection Systems (IDS) that detect well-known network attacks should be employed to further determine the vulnerability of the system. These tools should be used in an on-going manner. The system must continually be re-examined for vulnerabilities as information about newly discovered vulnerabilities is made available.

Section 5. Controls

A threat matched appropriately with a vulnerability creates a risk. A control is anything that may be used to work against either threats or vulnerabilities and thereby decrease risk. Examples of controls include computer passwords, door locks, and firewalls. Networked hosts should be secured to whatever extent possible by implementing appropriate controls. However, good network design and network device configuration can also provide a great amount of security for protecting networked hosts. This includes correctly placing and configuring firewalls, filters, and intrusion detection systems, as well as correctly configuring routers and switches. The combination of the network design and configuration, with a consistently applied host security configuration policy creates a control domain – an area where certain security controls are in place, offering a distinct level of protection against threat.

When considering the controls to use in a system, the mission must be aware that certain paths along the communication between a host and the spacecraft may already provide some controls, and other paths may not. Therefore, a mission may need to use different controls over different portions of a communication path. A mission may also choose to use end-to-end network control solutions combined with consistent strong security policies for remote access hosts and servers containing critical information.

5.1 Control Domains

Several categories of network "openness" can be defined in terms of the security controls in place and the security attributes, or protection, afforded traffic on that network due to those security controls. These categories represent "control domains." Definitions of the security attributes and the three domain definitions follow. These domain definitions may help a system designer categorize existing networks which may be used in a system, and determine what, if any, additional controls should be put in place.

5.1.1 Security Attributes

A security attribute is a type of protection that is afforded to information. Definitions of security attributes used in this risk assessment are:

- Access control Process of allowing or disallowing access based on many criteria, including but not limited to successful authentication. Access control protects against unauthorized use of a network or networked resource.
- Availability Methods to maintain availability prevent the denial of service through degradation of network services.
- Authentication Establishment of the identity of an entity (either a user or a computer system). Authentication in this document also includes non-repudiation.

- Non-repudiation Non-repudiation is the provision that an entity sending or receiving a communication cannot later deny sending or receiving that communication.
- Confidentiality Methods to maintain confidentiality protect information from being disclosed to unauthorized entities.
- Data integrity Methods to maintain data integrity detect (but do not necessarily prevent) the unauthorized modification or deletion of data.
- Traffic flow integrity Methods to maintain traffic flow integrity prevent the collection of sensitive information about the network through observation of network traffic characteristics. This includes gaining information about the network based on when traffic does or does not flow or based on packet headers being sent to, from, or within the network.

5.1.2 Open Network Domain

An Open Network path is one which has not been certified by NASA as a network that fulfills NPG 2810.1 MSN requirements, and is accessible by persons unaffiliated with mission operations. This includes modem connections over the public telephone system, into a network.

This control domain, through the details of its security architecture, will not guarantee any security attributes. Open NASA networks, university networks, and the Internet are examples of open networks.

5.1.3 Open Mission Network Domain

An Open Mission Network path is one that has been certified by NASA as fulfilling NPG 2810.1 requirements for MSN information. It connects to other networks that may not fulfill the NPG 2810.1 requirements for MSN information. It is protected from these other networks by a router with port and IP address filters, and optionally by any other part of a secure gateway, as described in the description of a Closed Mission Network (below). Open Mission Networks may permit the initiation of connections from the outside through the filtering router or firewall. It is therefore sheltered from the Open Networks, but is still more vulnerable than a Closed Mission Network.

This control domain, through the details of its security architecture, should provide some limited availability, authentication, and access control. The protected nature of hosts on the Open Mission Network, together with application and network layer error checking should also provide limited data integrity. The Open IP Operational Network (IONet), which currently provides IP connectivity for the purpose of communicating limited command (under special circumstances) and telemetry data, is an open mission network.

5.1.4 Closed Mission Network Domain

A Closed Mission Network path is one which has been certified by NASA as fulfilling NPG 2810.1 requirements for mission (MSN) information, and is connected only to an Open Mission Network via a secure gateway. Wireless devices cannot exist in Closed Mission Networks, because the wireless access may provide an entry point into the network that could bypass the gateway[6]. The secure gateway may include external and internal routers, stateful firewalls, authentication mechanisms, and other protection devices, which control the input and output of information between the Open Mission Network and the Closed Mission Network. The gateway must be capable of creating and maintaining a log or audit trail of all packets that flow between the protected Closed Mission Network and the Open Mission Network. The gateway must also maintain state, such that it can recognize a packet as a legitimate part of a communication. It must selectively allow access based on International Standards Organization (ISO) Open Systems Interconnection (OSI) layer 3 and 4 information, and only allow communications which are initiated by systems on the Closed Mission Network.

This control domain, through the details of its security architecture, should provide availability, confidentiality, authentication, and access control. The protected nature of hosts on the Closed Mission Network, together with application and network layer checks should also provide reasonable data integrity. The Closed IONet, which currently provides IP connectivity between MOCs and ground stations for the purpose of communicating command and telemetry data, is a closed mission network.

5.2 Current Controls

Some controls are already in place for the communication paths and computers that are used to communicate with and control a spacecraft. Many are mandated by NASA and Center policy. Missions that connect to either the Closed IONet or Open IONet must comply with the requirements and policies for that network as stated within the *Internet Protocol Operational Network (IONet) Access Protection Policy and Requirements*[7]. Other networks may have their own security policies to protect against threats. Each piece of a communications path used should be evaluated to determine what controls it already offers. Then mission management must decide if those controls are sufficient to meet the NASA requirements for the type of information communicated, and sufficient to appropriately mitigate the mission's risk. Ideally, a Closed Mission Network would require few, if any, additional controls. An Open Network may require the strongest set of controls. The appropriate level must be determined by the mission, as the mission must accept responsibility for determining and mitigating its own risk.

More details about the types of controls that may be applied to a system, and factors that affect the efficacy of the controls are given in Appendix C.

5.3 Controls Specific to IP

Many controls are not specific to any particular network protocol. IT resources can benefit greatly by physical access control and protection. In addition, plans to recover from physical or electronic damage can decrease recovery time and minimize the impact of an incident. These

[6] As security for wireless devices improves, their inclusion into Closed Mission Networks may be possible.

[7] Document 290-004 at http://forbin2.gsfc.nasa.gov/290%20iso/doc%20web%20page/document.htm

controls are independent of protocol. Controls such as IP Security (IPSec), TCPWrappers, firewalls, and most commercial Intrusion Detection Systems are more specific to the IP protocol. A layered approach to both physical and digital controls should be employed to mitigate risk.

Section 6. Risk and Mitigation

6.1 Risk Determination

The combination of threat, vulnerability, and current controls defines the risk. A typical risk assessment will prioritize the known risks and propose a plan for mitigating each unacceptable risk. Given that this risk assessment is not scrutinizing a specific IP-in-Space mission, generic threats, vulnerabilities, and controls were evaluated to determine and prioritize risk in the system. The risk analysis team determined what information required protection, and to what extent (this was a way of identifying and placing value on informational assets). The generic threat and vulnerabilities were all considered possible. Finally, the likelihood that the required protection would be afforded the information over different possible control domains (Section 5), which have inherent vulnerabilities and controls, was determined. The areas of greatest risk are those areas where the information was not likely to receive the desired protection against known threats and vulnerabilities.

6.2 Information Protection Requirements

6.2.1 Applicable Policy

Several Federal, Agency, and Center policies and recommendations impact the information protection requirements. Missions must, at a minimum, meet with the protection requirements in any of the applicable policies and guidelines. Some relevant policies and guidelines are listed:

- Computer Security Act of 1987
- OMB Circular A-130 Appendix III
- Clinton Policy on Critical Infrastructure Protection: PDD 63
- NIST Special Publication 800-23
- NPD/NPG 2810.10 IT Security
- National Security Telecommunications and Information Systems Security Committee National Security Telecommunications and Information Systems Security Policy (NSTISSP) Number 11, “National Policy Governing the Acquisition of Information Assurance (IA) and IA-Enabled Information Technology Products.”
- NTISSP 12
- National Security Directive 42
- International Traffic in Arms Regulations (ITAR)

6.2.2 Recommended Requirements

The following minimum information protection requirements shall be employed in IP-in-Space missions. Any mission may choose to implement controls to provide stronger protection than these minimum requirements if they feel their information and resources warrant it. Every mission should carefully scrutinize their information and resources and the applicable policies to determine the required level of protection for their information. Table 6-1 indicates a checkmark where a given security control, or protection, should be afforded a type of information.

Table 6-1: Recommended Information Protection Requirements

NPG 2810.1 Category→	**SER, PUB, ADM**	**BRT, MSN**	**MSN**
Data Categories → ***Security Controls*** **|** **V**	**Ordinary Downlink**	**Mission Downlink**	**Mission Uplink**
Availability	–	–	✓
Data Integrity	✓	✓	✓
Confidentiality	–	✓	✓
Authentication/ Non-Repudiation	✓	✓	✓
Access Control	–	✓	✓
Traffic Flow Integrity	–	–	–

6.2.2.1 Availability

Availability is required for transmitting all Mission Uplink data because of the time-dependent nature of much of this data. Especially in emergency situations, a command must be issued at a specific time, and it cannot arbitrarily be sent at a later time. Therefore, the network and networked devices must always be available for Mission Uplink.

Availability is also highly recommended for all real-time downlink communications when the real-time telemetry is not stored on the spacecraft. Anytime information that is not already safely archived is "downlinked," guaranteed availability should be provided. Availability is also recommended for any command acknowledgements or other spacecraft responses that may be needed right away.

6.2.2.2 Data Integrity

All NASA owned data, whether scientific, command, or health and status messages must be delivered as accurately as possible. Therefore, data integrity should be required for all communications.

6.2.2.3 Confidentiality

The minimum recommended confidentiality requirement is on Mission Downlink and Mission Uplink. If legitimate commands were visible to a hostile entity, that entity would be capable of sending the same command to the spacecraft, possibly causing great harm.

6.2.2.4 Authentication & Non-repudiation

Missions should be assured that data is being sent to and received by only the appropriate entities. In addition, they should ensure that any received data was actually sent by an appropriate entity. Authentication can be used to accomplish this. Therefore, on all communications, authentication is required. In this document, successful authentication includes non-repudiation.

6.2.2.5 Access Control

NASA should be assured that sensitive data is being accessed only by the appropriate entities (human or computer). Therefore, Mission Uplink and Mission Downlink must be protected by access control.

6.2.2.6 Traffic Flow Integrity

Traffic flow integrity is not a minimum requirement. Since, in most missions, entities will be able to tell when data will be sent or received simply by knowing the orbit of a satellite, which is easily attainable information. Furthermore, as long as replay attack prevention, confidentiality of commands, and strong authentication are maintained, an entity will be unlikely to make use of any information gleaned by compromising traffic flow integrity.

6.2.2.7 Requirements Scope

The information protection requirements should be met to the best of the ability of each mission. Different methods of varying cost and efficacy may be used to meet the requirements. However, controls to provide the required protection should be implemented to such a degree that the mission could be reasonably assured the security scheme would protect the assets appropriately. The extent to which these requirements are met or not met determines how well the risk for a mission is mitigated. Mission and NASA personnel responsible for the success of the mission must determine if the risk mitigation is appropriate for the mission. How well the requirements are met should be appropriate to the risk involved.

6.3 Control Domain Requirement Satisfaction

The control domains described in Section 5.1 offer varying levels of the protection needed to meet the suggested requirements.

Closed Mission Networks, by definition, provide data integrity, availability, confidentiality, authentication, and access control. If a system uses only Closed Mission Network resources to communicate with the spacecraft using IP, then it meets the requirements and will likely not need any additional security measures other than those necessary to secure the wireless link between the ground antenna and the spacecraft.

Open Mission Networks, by definition, provide limited data integrity, availability, authentication, and access control. Therefore, it is possible to receive Ordinary Downlink data from a spacecraft on an Open Mission network if appropriate authentication methods are employed.

On an Open Network, no security attributes are guaranteed. None of the information protection requirements can be satisfied without additional measures built into the system.

6.3.1 Qualification of Control Domains

A process should be created to qualify networks or communications paths as meeting the specifications for different control domains. This type of qualification would allow mission planners to know what protection is provided by networks that have been implemented for the mission. This will help determine what additional security measures must be designed into their systems.

Until a process is implemented to officially qualify networks and communication paths as fitting the criteria for one of the control domains, missions should rigorously evaluate the communications paths themselves. Each NASA network is currently required to have a risk assessment and security plan, as well as a process for reviewing and updating the security plan, procedures, and measures. The must also undergo penetration testing and periodic security audits. This proof of security may assist mission planners in their evaluations of networks.

6.4 Prioritized Risks

The risks inherent in IP-in-Space mission scenarios are listed in the sections below in prioritized order, with the greatest risk listed first.

6.4.1 Risk on Open Networks

The risks on Open Networks are primarily a result of the lack of guaranteed availability and capability to adequately secure a host outside a mission network environment controlled by NASA or a NASA partner facility. This risk to missions communicating sensitive information over open networks is unacceptable. Mitigation of such risk may not be possible at this time. It is generally recommended that Mission Uplink and Mission Downlink not be transmitted over open networks at this time. The risk of sending communications involving payload (i.e. instrument) command might be acceptably mitigated on open networks today, provided the instrument system can be appropriately isolated from the spacecraft system, and dependant upon the value of the instrument and the science data it produces. It is recommended that any such effort be conducted only after carefully analyzing the risks and integrating security into the design of the system from the initial stages of development. Any such effort should be treated as a prototype, and should be used to study possible security architectures and their success or failure.

6.4.1.1 Availability Disrupted

The greatest risk inherent in IP-in-Space missions is that the network availability fails on an Open Network while a command is sent. Availability cannot be adequately guaranteed on an open network such as the Internet. Denial of Service (DoS) attacks, network equipment failure, or congestion is likely. While the Internet does happen to have many redundant paths, single points of failure still exist, especially near communication endpoints, or at some service provider Points of Presence (POPs).

Potential Mitigation

The best any mission can do to mitigate the risk of unavailability on an Open Network is to provide a traditional Closed Mission Network control center with redundant connections to the ground station antenna(e) that may be used as a backup when Open Network connections fail. Even then, some communications may be lost in the time it takes to switch to the backup Closed Mission Network Control Center.

6.4.1.2 Host Vulnerability High

The risk of an individual networked host compromise is increased on open networks because of the connectivity to a greater number of potentially harmful entities, and because of the well-known nature of the IP protocol implementations and applications and their exploits.

Measures to provide Data Integrity, Confidentiality, Authentication, and Access Control exist, and may be built into a system using open networks to communicate with a spacecraft. However, these measures are likely to break down too easily when they are employed in an environment with insufficient resources for monitoring and checking control success. For instance, a Virtual Private Network (VPN) tunnel between a spacecraft and an Internet-connected host in a scientist's house may provide confidentiality, access control, and authentication. However, the host may be easily compromised by a physical intruder, or by a hacker entering the host through a virus, trojan, or other digital attack mechanism employed as the scientist surfs the web outside the VPN for the latest news or stock quotes. Once the host is compromised, a VPN may no longer provide proper confidentiality, access control, and authentication. Security is only as strong as its weakest component. The weakest component in this scenario is a host in a home which lacks resources to sufficiently electronically or physically secure the host, and which is out of the purview of NASA personnel who could ensure the physical and electronic security and integrity of the system.

Potential Mitigation

NIST- and NSA-certified security products are recommended for use (and are required for any encryption products). The overall security solution should be tested and approved prior to implementation. Finally, strict physical access controls should be employed at the host location. (See Appendix C for examples of useful controls.)

These security measures must be implemented properly to appropriately protect the assets. System administrator, security administrator, and user education is essential for users to understand and maintain the security measures involved in their communication with the spacecraft. A system of remotely monitoring the open network host health and safety, and periodic audits of remote host configuration may also increase the likelihood that the security controls will remain in place, and reduce the risk.

6.4.2 Risk on Open Mission Networks

Open Mission Networks, by definition, provide data integrity. Therefore, with appropriate authentication controls, Ordinary Downlink communications are not at risk on Open Mission Networks. Open Mission Networks border Open Networks, and thus the Internet. For this reason, they are more susceptible to dangers of the Internet, and are subject to the same risks as Open Networks, although to a lesser extent because of the controls that are in place.

Because Open Mission Networks are owned and operated by distinct known entities that can be reasonably defined, the risks that are difficult or impossible to mitigate in an Open Network may be successfully mitigated in an Open Mission Network. Therefore, it may be possible to securely command spacecraft from Open Mission Networks if additional security measures are designed into the spacecraft communications system to fulfill the information protection requirements.

Potential Mitigation

Availability may be assured on an Open Mission Network if mission planners obtain redundant paths for Mission Uplink in any Open Mission network they use. Mission planners should also take extra precautions to ensure confidentiality, authentication, and access control for Mission Uplink and Mission Downlink information. Possible measures include using IPSec, multiple forms of authentication, and strong physical and digital access control measures. (See Appendix C for examples of useful controls.)

6.4.3 Risks on the Ground Station to Spacecraft Link

Additional risks to communicating with a spacecraft using IP, over any of the control domains, are due to the last leg of the communication: the wireless uplink from a ground antenna to a spacecraft.

The wireless link is susceptible to eavesdropping, which may violate a confidentiality requirement. Further, since IP replay and spoofing attacks are better known than similar attacks in other protocols, the spacecraft may be left vulnerable to those attacks as well. Common IP attacks are less likely to occur over this wireless link than over the Open Network or Open Mission Network, because of the added expense and expertise required to purchase and set up an antenna and successfully target the wireless interface on the spacecraft.

Potential Mitigation

Proper use of IPSec may provide confidentiality, authentication, and data integrity on the wireless link. Another possible mitigation method is to allow the spacecraft to communicate only over certain areas, which have been physically secured to prevent placement of a hostile dish.

6.4.4 Risk on Closed Mission Networks

A Closed Mission Network, by definition, provides the required protection for all three types of data. However, risk is not eliminated entirely on a Closed Mission Network. The measures taken to secure a network such that it may be qualified as a Closed Mission Network must continually be monitored and improved. Security is a process that must continually be pursued.

The mission must actively participate in the program to monitor, maintain and when necessary, improve security.

6.5 IP-in-Space Security Handbook

A subsequent document entitled *IP-in-Space Security Handbook* will contain several possible solutions for mitigating risks and securely communicating with a spacecraft over the Internet. Several architectures involving standard IP security technologies are currently being studied and prototyped. The prototype results will be included in the handbook. Missions may use the information in the handbook to help determine the security architecture they need to employ in their mission. This handbook will provide proof-of-concept solutions. It will **not** provide a one-and-only solution for securing IP-in-Space communications, as such a solution is not possible. Any solution should include strong perimeter security, network and host-based security measures, intrusion detection and logging (with appropriate monitoring and reaction plans), strong authentication using multiple types of identifiers, and layers of diverse security measures, so that there is no single point of failure in the security scheme.

We recommend that the Handbook become a living document that NASA missions can add to and discuss. Any mission that investigates a security architecture or the application of a security technology in their mission may share the results in this living Handbook. The goal will be to enable NASA to better share its innovations and improve the posture of all missions by combining the forces of all personnel working toward the common goal of securing spacecraft communications over the Internet.

Appendix A. Threats

A.1 Intentional Human Threats

Intentional human threats are deliberate attacks by an individual against a computer system or network resource. Intentional threats must comprise both a motive and a means. Someone has to have a reason for their malicious intention, and that person must also have resources and knowledge sufficient to carry out their malicious intention. Some examples of the people who may harbor malicious intent and their motives behind it are as follows:

- Script kiddies or hackers want to gain bragging rights or resume material
- An employee is disgruntled, bitter, or bored and wants to make their day more interesting or wants revenge.
- A Corporation wants to gain a competitive advantage by learning information about a competitor or potential customer or by sabotaging a customer.
- Foreign nationals, or their agents, want to gain money or favor by learning information (government or defense related, economic, etc.).
- Foreign governments, or their agents, want to gain an advantage (monetary or strategic) by learning information (government or defense related, economic, etc.).
- Foreign nationals or foreign governments want to cause terror or destruction (physical or financial) in the name of patriotism, religious fervor, or old grudges.
- A national in disagreement with the government may want to gain information or cause terror or destruction in the name of some cause, religion, or belief.
- A thief wants to gain assets: either money, or things to use to get money.
- A thief wants to commit fraud to gain some assets.
- A curious student wants to push the limits to see what can and can’t be done and why – to see how things work.
- A paranoid person wants to make “them” stop spying on him or her.
- A mentally unstable person may have a myriad of reasons to cause disruption, many of, which are not foreseeable or understandable.

These people with potential motives to intentionally disrupt operations must also have means to carry out their intent. Some methods they can use to accomplish their goals are listed below. These attacks may be done over any type of link, or against any individual device in the system of communication between a spacecraft and a host, including the spacecraft itself.

- Corrupt, alter, or destroy information
- Destroy data or programs with logic bombs

- Enter data incorrectly
- Hold data hostage
- Crash systems
- Destroy or vandalize hardware or facilities
- Bomb threat or actual bomb
- Steal physical assets
- Steal information
- Use denial of service attacks
- Access facilities without authorization to accomplish one of the tasks listed above.
    - Social engineering
    - Forged badge
    - Stolen keys
    - Breaking and entering
- Access systems or information without authorization to accomplish one of the tasks listed above.
    - Social Engineering
    - Password attacks
    - Buffer Overflow attacks
    - Masquerade ("spoofing")
    - Eavesdropping
    - Forge digital keys or certificates
    - Session hijacking
    - format string attack
    - dictionary attack
    - resource subversion
    - recording and replaying authorized communications
    - other application, IP, TCP, or UDP attacks which exploit specific system vulnerabilities.

A.2 Unintentional Human Threats

Unintentional human threats are categorized as accidental or inadvertent acts that lead to the disclosure of information outside its intended and authorized audience. Unintentional acts may cause damage to computer and network resources or loss of processing capabilities due to the introduction of erroneous or incompatible data, omission of required data, and removal or erasure of required data. Such incidents may be the results of carelessness, ignorance, or lack of training. Some examples of unintentional threats include:

- Inadvertent disclosure
- Management error
- Human error
- Configuration errors
- Flaws in necessary procedures or processes such as:
    1. Configuration control
    2. Backup
    3. Incident handling
    4. Background investigation
    5. Auditing and logging
    6. Access control
    7. User identification and authentication

A.3 Environmental/Natural Threats

The Environmental/Natural threat category includes those natural phenomena and events of nature usually referred to as "Acts of God". Some of these include:

- Floods
- Earthquakes
- Lightning
- Electromagnetic interference
- Severe storms

A.4 Environmental/Fabricated Threats

The Environmental/Fabricated threats category consists of man- or machine-caused events that can have an impact on the availability of system and/or network assets. These may include:

- Power disturbance/outage
- Seepage/leakage
- Environmental/support system failure/malfunction
- Fire

- Hardware failure
- Software failure
- Communication failure/loss
- Substance abuse/employee impairment

Appendix B. Vulnerabilities

B.1 Vulnerability Categories

Vulnerabilities will generally fall into the following categories. When examining an IP-in-Space system, missions should consider the following types of possible vulnerabilities in their systems.

B.1.1 System Administration

System[8] and security administrators must continually maintain their systems to keep them as safe as possible. Errors in system configuration, or poor system security configuration are major sources of vulnerabilities that can be maliciously exploited. Many successful DoS attacks are actually self-inflicted and are due to any of the threats listed in Appendix A in conjunction with one or more of the following vulnerabilities:

- Operating system and software patches are not kept up-to-date.
- Unused ports are not disabled.
- Unused software is installed.
- Access control is weak (file sharing settings, passwords, trust relationships, etc.).
- Superfluous trusted hosts and accounts are permitted.
- Freeware administration tools are not reviewed properly and may have hidden malicious code.
- System administrators give away passwords to people posing as legitimate users, without demanding proof of identity or authorization to access.
- Systems and applications are incorrectly configured.

B.1.2 Passwords or Account Procedures

Attackers will often access systems through an old account or through an account that was easily compromised. The following flaws in password and user account procedures can make systems more vulnerable to this type of attack.

- Passwords are weak (dictionary words, social information like family names).
- Accounts are not deleted in a timely fashion.
- Periodic audits of account validity are not conducted.
- Use of reusable passwords, particularly if they are sent in the clear over a link that can be monitored.

B.1.3 Operating Procedures

[8] In this section, a system refers to an individual host.

Operating procedure flaws can lead to other vulnerabilities, such as the ones in sections B.1.1 and B.1.2. However, they can also hinder recovery from an attack. The following procedures are essential, and flaws in these procedures, or a lack of these procedures can increase a system's vulnerability.

- Backups
- Disaster recovery
- Configuration management
- Virus protection
- Personnel management
- Physical access control
- Risk management
- Disposal procedures

B.1.4 Application or Operating System Software

Application or operating system software can be flawed. Each piece of software should be examined to determine if it has any known flaws. If possible, the source code should be examined for potential security and reliability problems. Some ways in which software can create vulnerabilities in an IT resource are:

- Software is not developed with security in mind
- Trapdoors are built in
- Default installs leave security parameters open
- Software fails to include checks for inputs that would cause a buffer overrun or other unexpected results.

B.1.5 Gullible or Careless Users

Users can be the weakest link in any system. Computer security is new to many people, so they are likely to fail to enforce security as it should be enforced. Some ways users can create vulnerabilities in a system are:

- Give away passwords to people posing as executives or system administrators.
- Set weak passwords.
- Download freeware with malicious or flawed code.
- Connect their personal laptop, PDA or other device to the network.

B.1.6 Encryption

Encryption is generally a good protective technology. However, if implemented poorly, it can add vulnerability to a system. For instance, encryption schemes can create a Denial-of-Service situation if a key is lost or corrupt, and no mechanism exists for the endpoints of the communication to establish or agree on a new key. Encryption algorithms and keys should be chosen such that the encryption is not easily cracked. Great care must be taken to protect keys.

B.1.7 Network

All communications can be intercepted. The difficulty is often doing it at a reasonable cost and not being detected. The inexpensive and easy end of the scale is plugging a 10 or 100 Megabit tap into a hub in an unprotected phone closet. The expensive and difficult end of the scale is intercepting quantum-encrypted communications.

Some tools for capturing IP traffic are easy to come by and use, particularly for 10- and 100-Megabit Ethernet. Equipment to capture all packets on heavily used Gigabit Ethernet and OC-12 circuits is beyond the budget that most individuals would be willing to spend. Intercepting traffic is a goal of most of the people behind the intentional threats listed in Appendix A.1. Intercepting traffic does not directly put the assets at risk, but intercepting traffic is a requirement for an attacker who wishes to modify the content of any communications. Being able to intercept traffic makes it easier to spoof communications, launch denial of service attacks that work, identify implementation errors, and collect reusable passwords.

Any attacker who is able to modify communications potentially has the ability to command the spacecraft, and therefore threaten the entire mission.

Spoofed communications, except those used to hide the source of a DoS, are generally used to take advantage of a trust relationship. The ultimate spoofed communication link would be a direct link to the spacecraft. If the attacker is able to convince the spacecraft to trust the link, the entire mission is at risk.

B.1.8 Limited Resources

DoS attacks can take almost any form. They usually exploit flaws in the design of IP and IP systems, or are brute force resource consumption attacks. Systems are vulnerable to DoS attacks due to vulnerabilities documented elsewhere in this section, as well as limited IT resources.

B.2 Vulnerability Factors

The following factors can affect the extent of the system vulnerability due to the causes described in the previous section.

B.2.1 Network Connections

The more hosts or networks to which a system is directly connected, the more vulnerable that system is. Each host connected to a system is another node from which an attack can be launched. Each network connected to a system adds even more hosts from which an attack can be launched. Even if the network connected to the relevant system is “trusted,” that network may be compromised, or also be connected, in turn, to other networks that are not trusted. Every host on those other networks is another host from which an attack may be launched. Firewalls, IP filters, and strict connection policies can limit the connections to a host or system, thereby limiting the scope of possible attack sources and threats.

B.2.2 False Sense of Security

Use of a strong perimeter defense creates a sense of security. IT resources inside a strong perimeter may omit or lessen security to improve ease of use. In this case, when the perimeter is cracked, the IT resources inside it are much more vulnerable. VPNs also provide a false sense of security. Users believe that they are safe because their communications are encrypted. However, if a VPN end-point system is cracked, dangerous communication may be transmitted over that VPN link. Users should remain vigilant, and continue to implement layered security measures.

B.2.3 COTS Integration

The choice of COTS products integrated into a system can improve or diminish security posture. Often, using COTS products leads to homogeneous systems. A homogenous system is one in which most or all of the devices use the same OS or applications. COTS products may also be built to open standards due to market demand. This leads to many products sharing similar vulnerabilities. These factors make it much easier for a single attack to create extensive damage on many or all devices in a system.

COTS bundling may also increase system vulnerability. Bundling multiple packages together into a single product increases vulnerability if any products that are unused are not or cannot be uninstalled. These unused portions of the software may offer an opening into the system that can be exploited by an attacker.

Many COTS products are rushed to market too quickly, and while functional, include many security bugs. The COTS products in their first releases are more likely to have these security bugs. They are also more likely to lack robust support and therefore may not be able to quickly patch those bugs. These buggy products increase system vulnerability.

B.2.4 Education of Users

Users who have had security education are less likely to make a mistake that will lead to a security breach, and are more likely to accept additional process steps necessary to maintain security. The level of security education of the users of a system directly impacts a system’s vulnerability.

Appendix C. Controls

This appendix describes some possible controls to consider for each of the vulnerable points in a system (Section 4.0), and describes some factors in the success of controls.

C.1 Available Controls

C.1.5 Proof of NPG 2810.1 Compliance

Any IT resource (network or networked device) owner who claims his or her IT resource satisfies the requirements in the NPG 2810.1 should prove that fact through a documented risk assessment and security plan. The security plan should specify exactly which NPG 2810.1 requirements are met.[9] The security plan should also outline all measures taken to secure the network and/or networked devices to the level deemed necessary by the NPG 2810.1. In addition, a process must be in place to review and document all configuration changes from a security standpoint. The existence of a security plan and appropriate reviews of that plan and its implementation improve the entire security control scheme.

A mission should review this proof of compliance with NPG 2810.1 for any IT resources it plans on using to communicate with the spacecraft, and decide if the controls in place are sufficient for its needs. The mission should also create these documents for any new IT resources it plans to maintain itself. The NPG 2810.1 requires all NASA networks produce a risk assessment and security plan.

C.1.6 Physical Security

Physical security should be strictly maintained on mission networks. All mission network equipment should be behind locked doors, and only authorized necessary personnel should be granted access. Physical security can help maintain access control, and can help provide a measure of confidentiality. If persons cannot get to the equipment, the information on the equipment may remain confidential. However, any cables that run through public areas represent a possible point of entry and weakness in maintaining confidentiality. This is the case currently with the Closed and Open IONet.

Physical security should be applied not only to primary IT resources, but also to filing cabinets or backup media that contain sensitive information, or information which could help someone defeat the system security controls, such as design documents and IP address information.

C.1.7 Radio Frequency Protocols

The use of frequency hopping or spread spectrum may make attacks on Radio Frequency (RF) communications harder to accomplish.

[9] The NPG 2810.1 specifies five categories of information, and levies different requirements on each category.

C.1.8 Authentication Devices

Access control schemes are often used with authentication schemes. Successful authentication may be a prerequisite for obtaining access to a resource. Authentication may be based on something you have, something you know, or something you are. By using more than one of these "somethings," such as a keycard that you have and a password that you know, authentication may be made more reliable. Some authentication methods use encryption, and are subject to issues described in section C.1.12 and B.1.7.

C.1.8.1 Passwords

Strong passwords can be used to help prevent false authentication and subsequent access compromise. Strong passwords should not be words found in any language dictionary, should contain at least 8 characters, and should be a combination of upper case letters, lower case letters, special characters, and numbers. However, even strong passwords that are sent unencrypted over a network link can be captured and used later by an attacker.

One-time passwords are preferred to strong passwords when the password must be sent in the clear over an accessible link. One-time passwords are calculated by the endpoints based on shared secrets (which never are transmitted over the link) and cannot be reused. One-time passwords can help prevent various forms of replay attacks.

C.1.8.2 Smartcards

Smartcards are usually used in conjunction with a password. A smartcard will either hold a chip that is recognized and approved by a card reader, or generate a one-time password. The idea behind smartcards is to use both something you have and something you know to more strongly authenticate a user. A smartcard may be lost, but without the password it is useless. A password may be cracked, sniffed, or stolen, but without the smartcard, it is useless.

C.1.8.3 Biometrics

Biometrics is also a form of two-fold authentication. Users use a password (something you know) and a biometric signature, such as a fingerprint, facial image, iris scan, or voice print (something you are) to gain authorization. Biometric authentication implementations usually have a higher false negative failure rate than a false positive failure rate. That is, they fail to authenticate authorized users more often than they authenticate unauthorized users. However, fingerprints can be molded, and digital forms of the biometric data can be stolen in order to thwart these devices.

C.1.8.4 Digital Certificates

Digital certificates are structures digitally signed by one entity carrying the public key and associated information of another entity. There are several standards using digital certificates. The financial industry uses X.968 from the International Telecommunication Union (ITU). The Simple Public Key Infrastructure (SPKI)/Simple Distributed Security Infrastructure (SDSI) standard is a product of the SPKI Working Group[10] of the Internet Engineering Task Force IETF. The Pretty Good Privacy (PGP) is a product owned by Network Associates, Inc. and is being developed as an open standard for encrypting email by the IETF[11]. The Domain Name System SECurity extensions (DNSSEC) is now a product of the DNSEXT[12] Working Group of the IETF. The DNSEXT Working Group has assumed the RFCs and drafts of both the DNSSEC and DNSIND working groups.

X.509 Certificates are the most widely known digital certificates and are defined in the X.509 standard of the ITU. X.509 certificates have progressed from version 1 defined in 1988 to version 3 that allow the inclusion of user defined extensions.

Digital certificates can be used for authentication since only the holder of the private key can decrypt the certificate. They can be used to establish a symmetric key between two parties to enable secure communications as in Secure Sockets Layer (SSL).

C.1.9 Code Testing and Auditing

Code testing and auditing can promote successful authentication, access control, data integrity, and availability. Ensuring that the code is written well will limit the flaws that can be exploited to circumvent control measures. Code should be examined for flaws such as vulnerability to buffer overflows or format string errors. Code should be examined to see if it opens any network ports or services that are unnecessary. Code should be tested to determine that it is robust enough to maintain availability, and that it handles data properly and guards against corruption.

C.1.10 Packet Filtering

Packet filtering is another method of providing access control. It restricts access based on IP address, which is a subset of access control that a strong stateful firewall provides. An Open Mission network, by definition, is sheltered from Open Networks by a filtering router at a minimum. The packet filters should restrict access to as few addresses as possible, with as much granularity as possible. Packet filters, or similar access lists, can reduce the number of hosts that have access to a potential victim.

C.1.11 Stateful Firewall

A strong firewall will provide access control, and may add a limited amount of confidentiality.[13] The protected side of a firewall is referred to as the closed side, and the network protected by the firewall is referred to as the closed network. A stateful firewall is a filter with the addition of the ability to track the state of session–oriented connections. A strong firewall should follow these guidelines:

- The firewall should be an application firewall, capable of maintaining state and looking deeply into the packet to see application information.

[10] http://www.ietf.org/html.charters/spki-charter.html

[11] http://www.ietf.org/html.charters/openpgp-charter.html

[12] http://www.ietf.org/html.charters/dnsext-charter.html

[13] By prohibiting unauthorized access to information, a firewall may help maintain the confidentiality of that information. However, a firewall alone will not ensure confidentiality.

- All communication should be denied by default, with specific rules allowing only necessary connections to pass.
- The rules allowing traffic to pass should specify IP address and port number, at a minimum.
- The rules allowing traffic to pass should only accept connections initiated from the closed side.
- The rules allowing traffic to pass from a Closed Mission Network should only allow connections to an Open Mission Network – not an Open Network. This will limit the pool of resources that could attack through the firewall to a set of resources with greater protection that Open Networks.
- Proxy services should be used whenever possible.
- The firewall should support logging capability.

C.1.12 Encryption

Encryption is used in many protocols and tools to restrict access to and provide confidentiality for information. It is not a panacea, and increases costs and complexity. It may be impractical in some situations and illegal in others. However, many missions may find encryption a useful tool.

C.1.12.1 Key Management

Key management is essential to the successful implementation of any encryption technology. Compromised keys can destroy confidentiality, and loss of keys can cause an inability to communicate. A robust, secure key management system must be developed for any system using encryption.

C.1.12.2 IPSec

IPSec is the standard IP protocol used to create Virtual Private Networks (VPNs). The protocol creates an encrypted tunnel between two endpoints, which may be either hosts or gateways. IPSec can be configured in Authentication Header (AH) mode, to provide data integrity and authentication, or in ESP (Encapsulation Security Payload) mode, to provide confidentiality as well. IPSec is usually associated with encryption. IPSec may be configured to use any of a number of encryption algorithms and encryption key lengths. IPSec should be implemented extremely carefully so that it provides the necessary level of security. It has many configuration options and could easily be configured in such a way that it does not provide the security intended. IPSec is typically used to protect all traffic between a system and any remote location.

C.1.12.3 SSL and TLS

Secure Socket Layer (SSL) and Transport Layer Security (TLS) provide server and client authentication and encryption for a single network connection. Authentication is optional and is accomplished by using digital certificates. Encryption may use a number of algorithms and key lengths, depending on implementation. SSL/TLS is typically used to protect only a system's connections with sensitive traffic. Other connections use unencrypted protocols. If only a portion of communications between two hosts required encryption, SSL/TLS may be preferred to IPSec for performance reasons. With IPSec, all communications between 2 endpoints is encrypted, but SSL and TLS can selectively encrypt only certain streams of a communication, based on port or socket information.

C.1.13 Intrusion Detection Systems

Intrusion Detection Systems (IDS) do not actively provide any security. Instead, they detect when the security controls have been defeated, so that appropriate corrective action may be taken. Without successfully monitoring the security in a system, there is no way of knowing if that security is working. Monitoring systems like IDSs are essential to properly maintain any security architecture.

C.2 Control Success Factors

Security controls may be implemented in varying degrees. Some factors that contribute to the effectiveness of a security control scheme, or security policy, are described in this section. These factors should be considered when designing the system.

C.2.1 Security Requirements in IT Resources Specification

Security must be an integral part of the system design. Security planning should be done in conjunction with reliability planning. Many security issues involve reliability, and all reliability issues affect security. Security requirements must be specified along with every other system requirement, so that the system designers may ensure that security is met. Security measures added on to a system after the fact are usually much less effective. **Including security in the design of a system is the single most important factor affecting the success of any set of security controls.**

In addition, security requirements should be specified when pieces of the system are purchased. Only products that meet or exceed the security specifications should be purchased for inclusion in the system.

C.2.2 Connections

As discussed in Section 4.2, the more connections a host has to other devices, the more vulnerable it is. Controls such as firewalls, IP filters, and strict connection policies can limit the connections to a host or system, thereby limiting the scope of possible attack sources and threats. The design of the system should limit the points of ingress and egress to other systems and networks. Those ingress and egress points should be protected and monitored.

C.2.3 Complexity and Limited Resources

Security measures tend to add extra complexity and require more resources. Since users do not like to add steps to their processes, and since resources can be scarce and expensive, the necessary security measures are often not implemented. This leaves the system more vulnerable.

C.2.4 Security Measure Layering & Diversity

In many cases, system users rely on one or a few strong security controls, such as a VPN or firewall system. However, it must never be assumed that because of a specific control, all risk has been eliminated. Because any given control may be compromised at some point in the future, systems should be built with layers of security. For example, network-based measures such as firewalls and filtering routers should be used with host-based measures such as TCP Wrappers. This will ensure that if an attacker gets past one countermeasure, another will be in place as an additional line of defense.

In addition, choosing diverse COTS products for each security measure (e.g., use two firewalls, each from a different vendor) to further layer security can prevent a single type of exploit from circumventing the entire security control scheme. COTS products that are supported by frequent patch support, and that are mature (are in a later release) are more likely to be more secure, and to recover security quickly when a security bug is found. Mature products should be used whenever possible.

C.2.5 Planned Responses

Disaster recovery planning is essential. Once an attack happens, a system should be in a position to respond to the attack, and restore functionality as soon as possible. Quick action can minimize the impact of an attack. Also, pre-planned smart action may preserve attack evidence and lead to a criminal conviction. If a response plan is in place, damage from an attack can be minimized.

Pre-planning can hurt security posture as well. Many security products can be configured to automatically make changes to routers or firewalls when attack signatures are detected. If the change made in response to attack signature detection is to cut off access to some service, then an attacker can more easily create a DoS attack by simply sending attack signatures. This type of preplanning may actually work against the success of security controls.

C.2.6 Coordination of Security Activities

Sharing security knowledge, information, and strategy, can better protect an entire organization. The security of a system is only as strong as its weakest link. When the organizations maintaining the security posture of a system work together to keep each other up to date and informed, security on the entire system is improved. Patches may be kept more up to date, and distributed attacks and other attack patterns may be seen and defended against.

C.2.7 User Education and Cooperation

Many controls rely on users of the system. Users must be educated about what system security controls they have a responsibility to maintain. Users must be educated about how to implement the security control properly, what the consequence of implementing the control incorrectly is, and what to do if an incident is suspected. Without user cooperation, security controls may be implemented so poorly that they are ineffective.

C.2.8 Level of Security Standards

A lack of standards can prevent sufficient protection. With no standards, organizations may adopt widely varying security architectures, which may be strong or weak. With standards of protection (standards do not necessarily imply homogeneity), each organization responsible for a piece of the security for a system will be sure to implement the minimum required level of security.

C.2.9 Assessment Capability

If security cannot be measured, it is impossible to determine if the security measures are working. An organization must be able to monitor its security to determine when it needs to be augmented. Tools such as IDSs, vulnerability scanners, and password crackers may be used to test a systems security posture. Without good assessment capability, good security cannot be employed.

Appendix D. Abbreviations and Acronyms

ADM	Administrative
AH	Authentication Header
BRT	Business and Restricted Technology
COTS	Commercial Off The Shelf
CSO	Computer Security Officer
DNSSEC	Domain Name System SECurity
DoS	Denial of Service
ESP	Encapsulation Security Payload
FOT	Flight Operations Team
GSFC	Goddard Space Flight Center
IDS	Intrusion Detection Systems
IETF	Internet Engineering Task Force
IONet	Internet Protocol Operational Network
IP	Internet Protocol
IPSec	Internet Protocol Security
ISL	Inter-Satellite Link
ISO	International Standards Organization
IT	Information Technology
ITAR	International Traffic in Arms Regulations
ITU	International Telecommunication
MOC	Mission Operations Center
MSN	Mission
NASA	National Aeronautics & Space Administration
NIST	National Institute of Standards and Technology
NPG	NASA Policy Guideline
NSA	National Security Agency
NSTISSP	National Security Telecommunications and Information Systems Security Policy

OMB	Office of Management and Budget
OS	Operating System
OSI	Open Systems Interconnection
PDA	Personal Data Assistant
PDD	Presidential Decision Directives
PGP	Pretty Good Privacy
PI	Principle Investigator
PUB	Public
RF	Radio Frequency
SDSI	Simple Distributed Security Infrastructure
SER	Scientific and Engineering Research
SPKI	Simple Public Key Infrastructure
SSL	Secure Socket Layer
TCP	Transmission Control Protocol
TDRS	Tracking and Data Relay Satellite
TLS	Transport Layer Security
UDP	User Datagram Protocol
VPN	Virtual Private Network

National Aeronautics and Space Administration

Goddard Space Flight Center

Code 290, Code 580

IP-in-Space Security Technology Solutions Recommendations & Prototype Effort

September 2001

Authored By:

Computer Sciences Corporation

GSFC

Under Contract:

S-38657-G TOR 16

Contents

List of Figures

List of Tables

Section 1. Introduction

Space Operations Management Office (SOMO), in cooperation with Code 291, has tasked Computer Sciences Corporation (CSC) under the Millennia contract S-38657-G TOR 16 to create a Security Handbook with recommendations for securely operating missions as nodes on the Internet. CSC worked with the OMNI project team, and a consortium of other Goddard Space Flight Center (GSFC) personnel and contractors to investigate technologies necessary for security solutions.

This document outlines the proposed solutions that were tested, and provides the results of the tests, along with recommendations for future efforts to help find the more secure solutions for using IP and the Internet to communicate with spacecraft. The findings lay some of the ground work for what must be a continuing effort by NASA to create, test, and refine IT security measures to assist in mitigating the risk involved in communicating with a spacecraft over IP on open networks.

All organizations involved in controlling spacecraft should, with due diligence, create, implement and enforce written security policies. The technologies discussed in this document are only a piece of a well-balanced, layered security policy.

Section 2. Proposed Solutions

Four likely routes for IP (Internet Protocol) communications with a spacecraft were chosen. These are presented as spacecraft operations concepts. Solutions focused on these as the most likely scenarios that NASA will need to protect. The solutions proposed for each of the four operations concepts are based on the fulfillment of recommended information protection requirements outlined in the document *IP in Space Risk Assessment and Information Protection Requirements.*

2.1 Network Definitions

The operations concepts use the following terminology to characterize networks along a communications route.

2.1.1 Closed Mission Network

A Closed Mission Network (CMN) is one that has been certified by NASA as meeting NPG 2810.1 requirements for mission (MSN) information, connected to only an open mission network, and connected to that open mission network via a secure gateway. Wireless devices cannot exist in closed mission networks, because wireless access may provide an entry point into the network that could bypass the gateway. The gateway may be a system including external and internal routers, firewalls, authentication mechanisms, and other protection devices that control the input and output of information between the open mission network and the closed mission network. The secure gateway must be capable of creating and maintaining a log or audit trail of all packet flows between the protected closed mission network and the open mission network. The gateway must also maintain state, in order to recognize a packet as a legitimate part of a communication. It must selectively allow access based on International Standards Organization (ISO) Open Systems Interconnection (OSI) layer 3 and 4 information, and only allow access as a response to an outgoing communication from the closed mission network and allow no incoming connections.

2.1.2 Open Mission Network

An Open Mission Network (OMN) is one that has been certified by NASA as meeting NPG 2810.1 requirements for MSN information, but it connects to other networks that may not fulfill the NPG 2810.1 requirements for MSN information. It is protected from these other networks by a router or device with port and IP address filters. Incoming connections are permitted. It is therefore sheltered from the open networks, but is more vulnerable than if it were protected by such a strong firewall.

2.1.3 Open Network

An Open Network (ON) is one that has not been certified by NASA as a network that meets NPG 2810.1 MSN requirements, and is accessible by persons unaffiliated with mission operations. A subset of these are modem connections – dial-in connections, over the public telephone system, into a network.

2.2 Requirements & Technologies

The *IP in Space Risk Assessment & Information Protection Recommendations* document outlines the minimum recommended protection for various types of communication with the spacecraft. These recommendations are summarized in Table 2-1. A check mark indicates that the protection should be afforded the type of data or information. Solutions tested were designed to meet these requirements.

Table 2-1. Recommended Minimum Information Protection Requirements

NPG 2810.1 Category→	SER, PUB, ADM	BRT, MSN	MSN
Data Categories → ***Security Controls*** **↓**	Ordinary Downlink	Mission Downlink	Mission Uplink
Availability (Av)	–	–	✓
Data Integrity (DI)	✓	✓	✓
Confidentiality (C)	–	✓	✓
Authentication / Non-Repudiation (Au)	✓	✓	✓
Access Control (AC)	–	✓	✓
Traffic Flow Integrity (TFI)	–	–	–

Security technologies were studied to determine which might provide the required protection. Table 2-2 includes some technologies that provide the necessary security controls. This is not an exhaustive list of technologies capable of providing the necessary protections.

Table 2-2. Security Control / Protection Matrix

Control Technology or Control Domain	**Security Protection Provided**						
	Availability (Av)	Data Integrity (DI)	Confidentiality (C)	Authentication (Au)	Access Control (AC)	Traffic Flow Integrity (TFI)	Notes
Firewall					✓		
IPSec (AH)		✓		✓	*		*Depending on the configuration, IPSec may provide access control.
IPSec (ESP)		✓	✓	✓	*	*	*Some TFI in Tunnel Mode. Depending on the configuration, IPSec may provide access control.
TLS		✓	✓	✓			
Kerberos				✓			
Hardening Scripts	✓				✓		
SSH		✓	✓	✓			
IDS							Good overall security measure, does not prevent attack.
Key Management							Integral piece of encryption and authentication
Smartcard				✓			
Frequency Hopping, Spread Spectrum	✓						
Biometrics				✓			
Closed Mission Network (CMN)	✓	✓	✓*	✓	✓		* Entities in the CMN who can access network traffic could see the data. It is assumed that entities in the CMN are trusted.
Open Mission Network (OMN)	*	*		*	*		* Less than CMN but more than ON.
Open Network (ON)							

2.3 Solution Concept 1: Communication Over Closed Mission Networks

Operations concept 1 is the traditional approach to communicating with a spacecraft. A workstation on a closed mission network, such as the Closed IONet, sends commands to, and receives data from the spacecraft (see Figure 2-1).

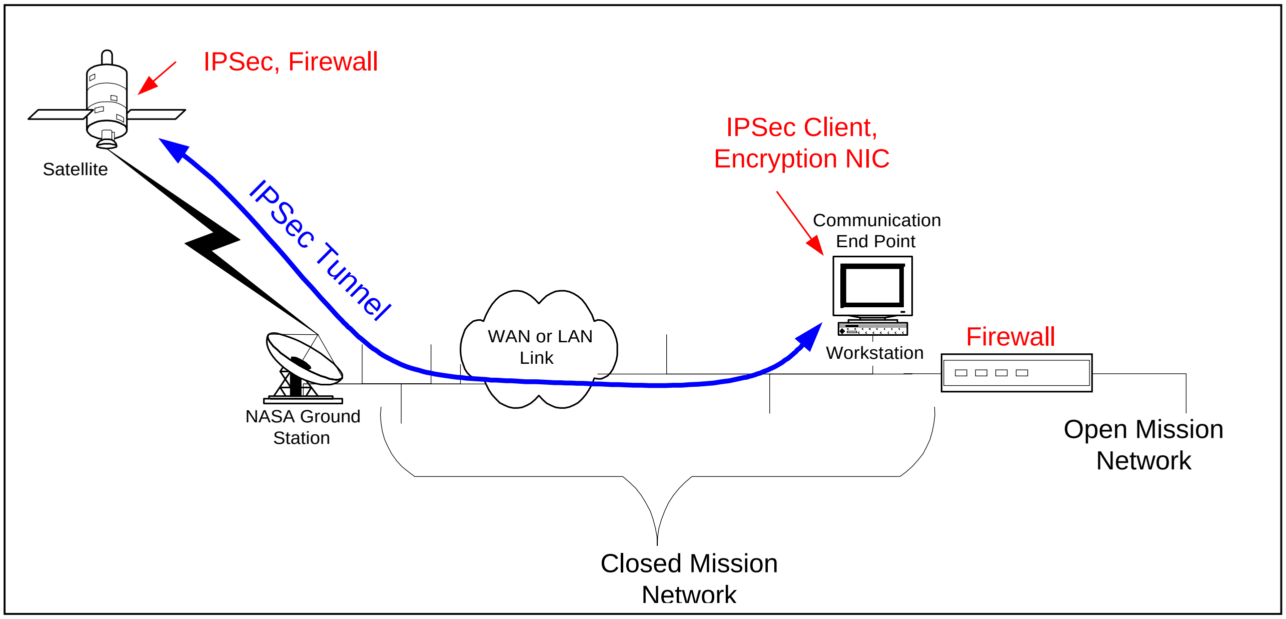


Figure 2-1. Communication over Closed Mission Networks

Many missions currently use IP to carry communications within the bounds of closed mission networks, as shown in this concept. However, the last path of this communication route to the spacecraft, the RF (Radio Frequency) link between the ground station and the spacecraft, currently does not use IP. To use IP in this path may require additional security measures. Because this RF link is common to all IP-in-Space scenarios, testing this concept has the highest priority in the study. The solutions proposed include implementing an IPSec (IP Security)[1] VPN (Virtual Private Network) end-to-end, with a firewall on-board the spacecraft. Another potential solution for this scenario is to provide a VPN gateway at the ground station, to encrypt only the RF link. This is included in concept 2 and will be addressed in concept 2 testing.

IPSec VPN tunnels were tested to determine the feasibility of encrypting communications between a spacecraft and a ground host. Encryption and/or IPSec were tested on both BlueCat Linux and OpenBSD operating systems running on 486 PCs to simulate a spacecraft. IPChains were tested as the on-board firewall on the BlueCat Linux system.

Table 2-3 shows the security that this possible solution would offer the spacecraft and data.

[1] See Appendix B for an IPSec overview.

Table 2-3. Security Provided by Solution Concept 1

	Security Controls Provided						
Technology or Control Domain	Av	DI	C	Au	AC	TFI	Notes
With data on the Closed Mission Network (including commanding host)							
CMN	✓	✓	✓*	✓	✓		*Assumes entities in CMN are trusted.
IPSec (ESP)		✓	✓	✓	*	*	*Some TFI in Tunnel Mode. Depending on configuration, may provide access control.
With data on the RF Link Path (including spacecraft)							
IP Chains Firewall					✓		
IPSec (ESP)		✓	✓	✓	*	*	*Some TFI in Tunnel Mode. Depending on configuration, may provide access control.

2.4 Solution Concept 2: Communication Over Open Mission Networks

Communicating over an Open Mission Network through a Closed Mission Network to the spacecraft is similar to communicating over the Open IP Operational Network (IONet) through the Closed IONet. See Figure 2-2.

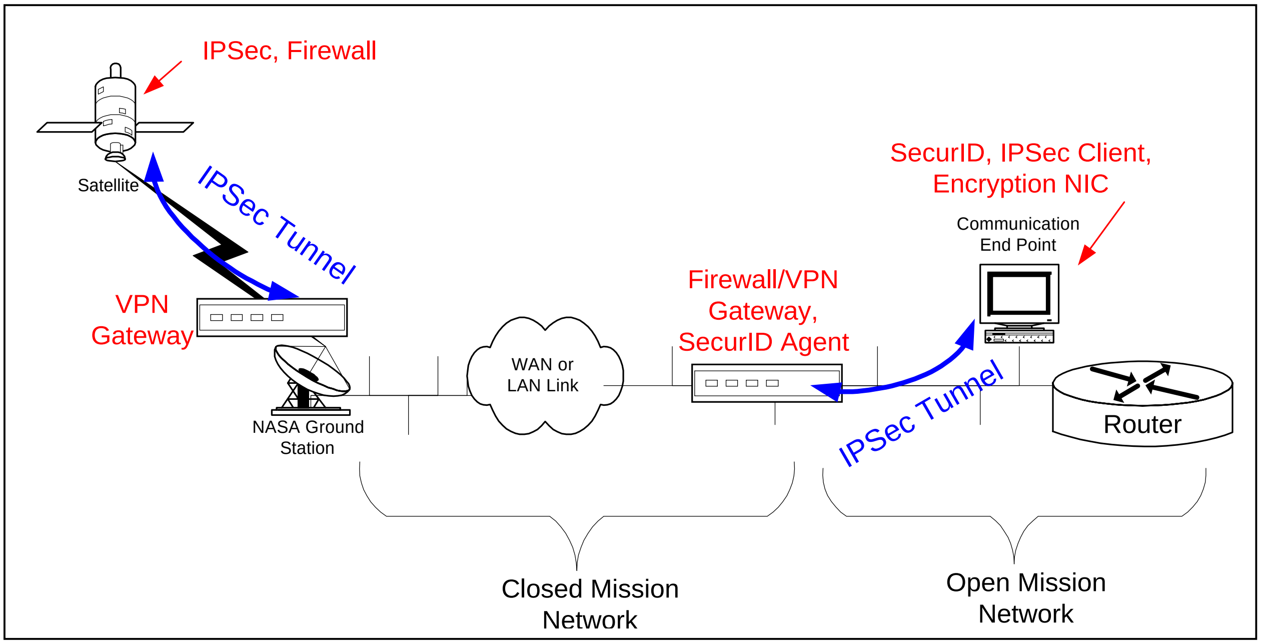


Figure 2-2. Communication over Open Mission Networks

Commanding over an Open Mission Network requires additional security measures. The major difference in the approach of concepts 1 and 2 is that the control center host is less trusted in concept 2 because it may not have strong host protection, and because it resides on a more open network. Therefore, concept 2 requires much stronger authentication, additional measures to prevent the ground host from being compromised, and measures to detect any compromise that may have occurred. As in concept 1, IPSec provides authentication, confidentiality, and data integrity. The firewall provides access control. SecurID provides a secondary method of authentication.

NICs that can implement encryption algorithms in hardware will be tested in Windows ground hosts to determine to what extent offloading the encryption to hardware improves performance of the IPSec VPN tunnel. This solution contains two firewalls. One is the firewall protecting the Closed Mission Network from the Open Mission Network, the other is the firewall on-board the spacecraft. A Cisco PIX 515 firewall will be used to simulate the firewall between the Closed Mission Network and other networks. A Netscreen-5 firewall will be used as the VPN gateway at the ground station.

A host on the Open Mission Network should already comply with the NPG 2810.1 MSN information requirements. However, the host should be hardened as much as possible, perhaps using tools such as host-based firewall, host-based intrusion detection system, local monitoring, and remote monitoring on a centralized log server. These methods of ensuring the security of a host system were not tested due to the extreme variability of possible configurations.

Strong authentication, including non-repudiation, is crucial when the ground host resides on an Open Mission network. Authentication using only weak methods, such as username and reusable password is not acceptable. SecurID cards or other smartcards should be used as a secondary authentication measure. The smartcard server would likely reside behind the firewall to the Closed Mission Network. The IPSec tests conducted for concept 1 are sufficient for the spacecraft portion of IPSec in concept 2. However, running a VPN through the firewall must be tested. There are two options: one is to allow the entire IPSec tunnel to pass through the firewall directly, terminating at the spacecraft, and the other is to use a firewall with VPN capabilities as a gateway to the Closed

Mission Network. The first option has the disadvantage that it likely means secondary authentication using a Smartcard or SecurID card would require even more software on the spacecraft. Additionally, we do not recommend that an encrypted tunnel ever be permitted through a firewall. Doing so limits the control over what exactly passes through the firewall, and opens a hole to all IP ports and protocols.

Another option is to use a firewall with VPN capabilities in the place of the Closed Mission Network Firewall. In this scenario, the authorized traffic would be encrypted outside the firewall, decrypted at the firewall, inspected by the firewall, and transmitted in the clear over the Closed Mission Network. In order to secure the communications for transmission over the RF link, another VPN gateway would be used at the ground station, or a proxy server for the communications, inside the Closed Mission Network, could initiate an IPSec VPN tunnel to the spacecraft. Allowing the Closed Mission Network firewall to inspect unencrypted packets by IP address, and port, protocol, or other layers, is preferred.

To evaluate this option, a double IPSec tunnel was tested: one tunnel from the host to the firewall, and another from the groundstation to the spacecraft.

Table 2-4 shows the security that such a solution would offer the spacecraft and data.

Table 2-4. Security Provided by Solution Concept 2

	Security Controls Provided						
Control	Av	DI	C	Au	AC	TFI	Notes
With data on Open Mission Network (including commanding host)							
Host Hardening					✓		Strong authentication and access control measures necessary to provide AC at the host.
IPSec (ESP)		✓	✓	✓	*	*	*Some TFI in Tunnel Mode Depending on configuration, may provide access control.
Open Mission Network	*	*		*	*		* Less than CMN but more than ON.
With data on Closed Mission Network							
Firewall					✓		
Closed Mission Network	✓	✓	✓*	✓	✓		*Assumes entities in CMN are trusted.
With data on RF link (including spacecraft)							
IPChains Firewall					✓		
IPSec (ESP)		✓	✓	✓	*	*	*Some TFI in Tunnel Mode. Depending on configuration, may provide access control.

Availability is not guaranteed over the OMN portion of the communications path. Host hardening should increase the likelihood of meeting the availability requirement by ensuring that security and OS patches are constantly kept up-to-date. A host-based firewall or IDS system may also increase system availability. The protection afforded by the Open Mission Network filtering routers also increases system availability. Each mission must address in a risk assessment the measures taken to ensure availability on this portion of the communication path.

Availability is also not guaranteed across the wireless (RF) portion of the communications path. Having multiple ground stations or relay satellites that can transmit and receive from the spacecraft increases the probability of availability, as does multiple receivers and transmitters on-board. However, availability cannot be wholly guaranteed. This is no different from the case today. The extent that availability is guaranteed by redundancy and contingency measures must be addressed in the mission's risk assessment.

2.5 Solution Concept 3: Communication over an Open Network

This concept builds further on the previous concepts. The above solution concepts will address all paths in this concept except the open networks between the Open Mission Network and the host sending communications.

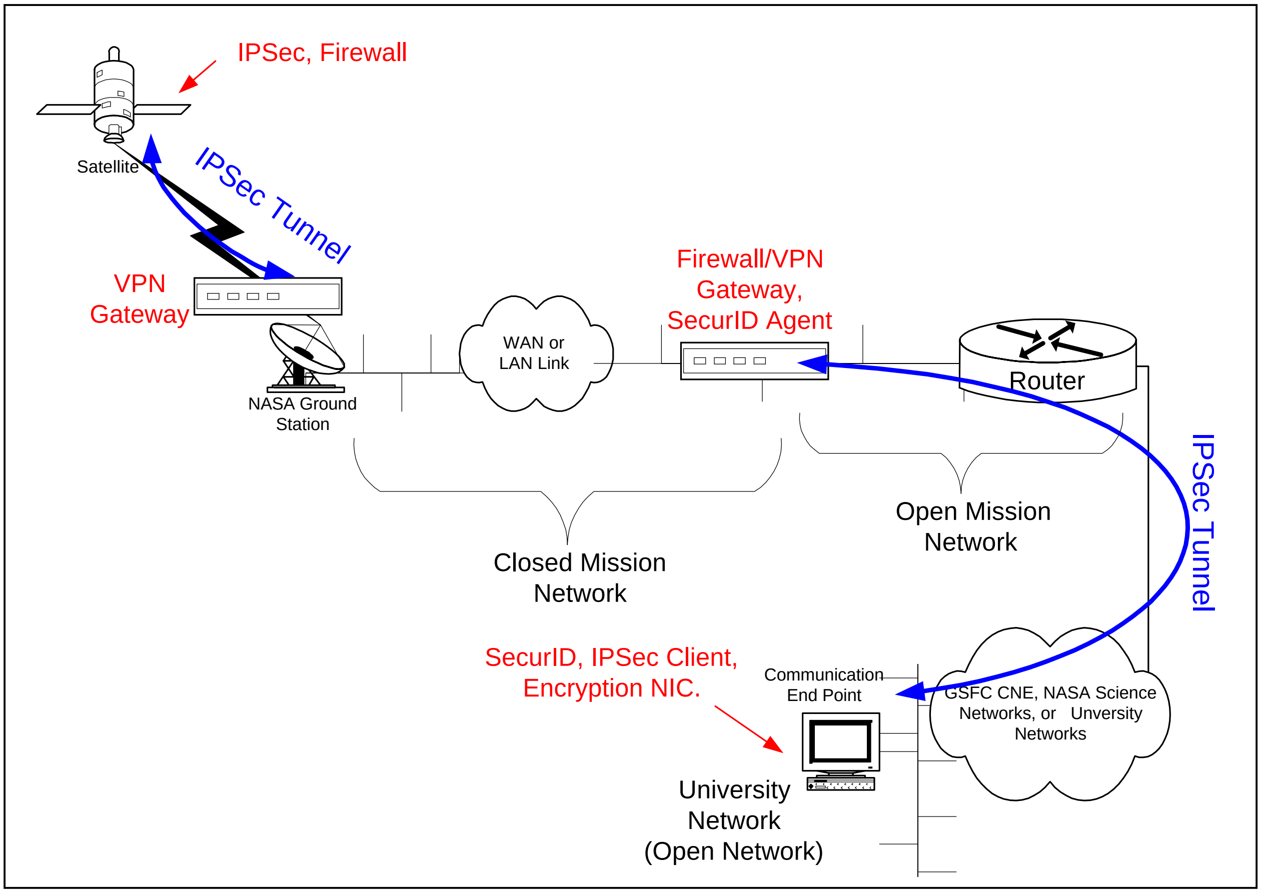


Figure 2-3. Communication over Open Networks

This concept differs from concept 2 in that persons who are not affiliated with the mission can more easily intercept traffic flowing on the open networks. Because NASA often coordinates with universities and other scientific partners, this is a likely scenario for IP-in-Space missions. It is considered the third priority in the study.

The solution for this concept is the same as the solution for concept 2. However, in this case, host security standards, along with methods for monitoring the integrity of the ground host become even more important. There are too many and too varied host security measures to explore in the scope of this effort. Therefore, testing will be the same for both concept 2 and concept 3.

Table 2-4 indicates the proposed security architecture provides the recommended protection for the sensitive mission data over the RF Link, the Closed Mission Network, and the Open Mission Network. The following table shows that the proposed security architecture may also provide the recommended protection over the Open Network portion of the communications path if sufficient controls are in place to protect against a compromise on the commanding host.

We emphasize that it is quite difficult to determine "sufficient" controls to protect the host, and it is even more difficult to keep a sufficient level of controls current with the rapid pace of new IT attacks.

A mission must address in its risk assessment the extent a host on an Open Network is protected

Table 2-5. Security Provided by Solution Concept 3

	Security Controls Provided						
Control	Av	DI	C	Au	AC	TFI	Notes
With data on Open Network (including commanding host)							
Host Hardening					✓*		*Strong authentication and access control measures necessary to provide sufficient AC at the host.
IPSec (ESP)		✓	✓	✓	*	*	*Some TFI in Tunnel Mode. Depending on configuration, may provide Access Control.

2.6 Solution Concept 4: Communication Over the Internet

This last concept is the vision for the future of IP-in-Space. In this concept, either the spacecraft, or some portion of the spacecraft, or some information is owned by or affiliated with NASA. However, all of the intermediate networks, including the ground station are open networks. This scenario may include multiple support routes. Missions that require large amounts of data on a downlink and/or multiple communication contacts per orbit may need to use both NASA and non-NASA ground stations. Mission formulators need to understand that different security policies may apply to their mission depending on which support infrastructure is being used. NASA may restrict the use of their assets depending upon the risks associated with the mission requesting support.

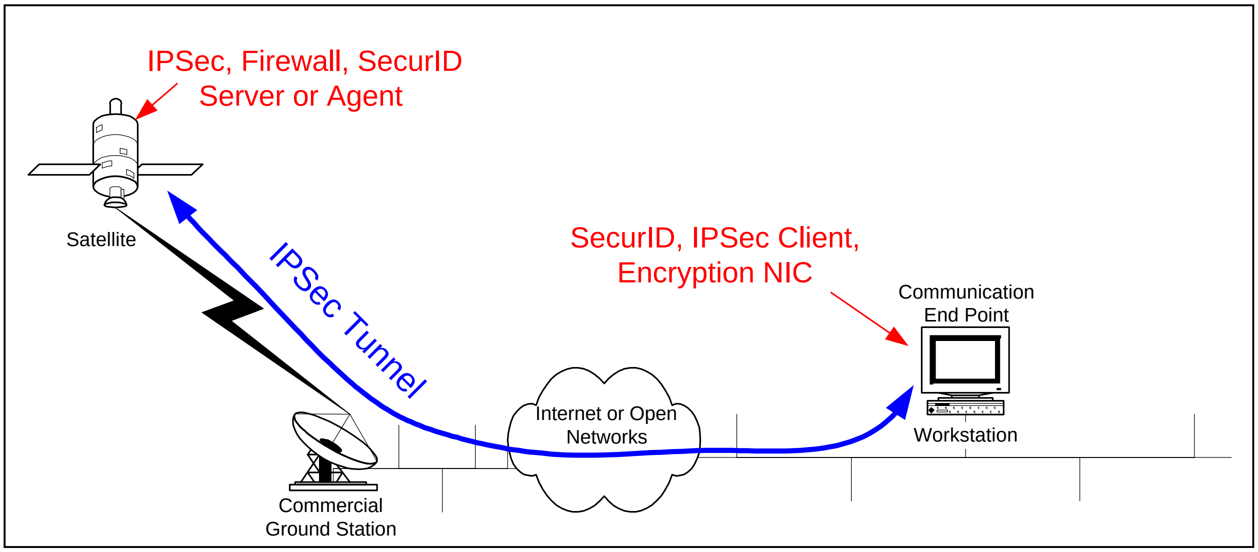


Figure 2-4. Communication over the Internet

This is the riskiest concept, and will require a multifaceted, layered security solution. This concept requires the same rigorous host-based security measures as in concept 3. The host should also participate in some form of monitoring that can help detect whether or not the system has been compromised. For better authentication, smartcards are required, and smartcard components, such as a server or agent, must be on the spacecraft.

Again, we emphasize that it is quite difficult to determine "sufficient" controls to protect the host, and it is even more difficult to keep a sufficient level of controls current with the rapid pace of new IT attacks.

A mission must address in its risk assessment the extent a host on an Open Network is protected

Table 2-6. Security Provided by Solution Concept 4

Control	**Security Controls Provided**						
	Av	DI	C	Au	AC	TFI	Notes
With data on Open Network (including commanding host)							
Host Hardening					✓		Strong authentication and access control measures necessary to provide sufficient AC at the host.
IPSec (ESP)		✓	✓	✓	*	*	*Some TFI in Tunnel Mode. Depending on configuration, may provide access control.
With data on RF link (including the spacecraft)							
IPSec (ESP)		✓	✓	✓	*	*	*Some TFI in Tunnel Mode. Depending on configuration, may provide access control.
IPChains Firewall					✓		

Section 3. Test Results Summary

3.1 Test Measurements

Many testing efforts were conducted in several different labs. Whenever possible, testing observations were focused on the technologies. This testing was not meant to differentiate between vendors' implementations of the technologies or to endorse any one vendor over any other. The purpose was to determine if a solution with a mix of technologies and tools could be built to allow the necessary spacecraft communication and provide security. Each concept was configured so that traffic could flow from the simulated control center host to the simulated spacecraft, through the appropriate network paths, and with the appropriate security technologies in place. These scenarios were set up as a proof-of-concept to show that the technologies proposed could be configured to work together as indicated in the concepts in section 2. To the greatest extent possible, certain common elements were measured in each effort. These elements are:

- CPU and/or Memory Utilization
- Latency or Delay
- Bandwidth utilization and/or overhead
- Complexity or Ease of Use and Interoperability (subjective measure)
- Security Features
  - Notes on the protection the technology provides and to what extent
  - Whether or not the technology is NIST/NSA NIAP certified or compliant with other standards

3.2 Results Summary

Solution concepts 1-3 were successfully created in lab environments. Solution concept 4 could not be completed because access to the SecurID agent software necessary for BlueCat Linux could not be acquired. Therefore, strong authentication at the spacecraft was not tested. Concept 2 was the most complicated concept, but the concept most likely to occur. VPNs through the Open IONet are already occurring, and adding encryption on the uplink would be appropriate for the first projects that use IP to communicate with a spacecraft. In addition, in the first instances of IP in Space, proxy command centers may be used on the Closed IONet for checking commands before uplink.

3.2.1 CPU Utilization

Impact of encryption on CPU resources was evident in these tests. However, the extent of the impact varied, and no hard and fast formula emerged for determining the precise impact of encryption on CPU resources. The design and implementation of the security products and the system using these products will affect the CPU resources needed to support the products. In addition, the applications used to transport data may affect the CPU utilization. Therefore, it is difficult to predict generically but accurately how adding security for IP in Space will affect the CPU resources. Mission developers must consider what technologies or products will be included in a mission to add the necessary security. They must then design those technologies or products into the system so that CPU resources are available for all the necessary processes. Figures 3-1 and 3-2 show the impact on CPU utilization when IPSec is employed. FTP was used to transfer data. In the graphs, the baseline values do not include encryption, the scenario 1 and scenarios 2 & 3 values refer to the values obtained with IPSec, as indicated solution concepts 1, 2, and 3 discussed in section 2 of this document. Figure 3-1 shows CPU utilization for a 486 25 MHz PC with the OpenBSD operating system. Figure 3-2 shows CPU Utilization for a slightly more powerful 486 66 MHz PC running BlueCat Linux. Clearly both hosts are stressed by encryption function. The more powerful BlueCat host, however, is able to handle the load half of the time without using all its CPU resources.

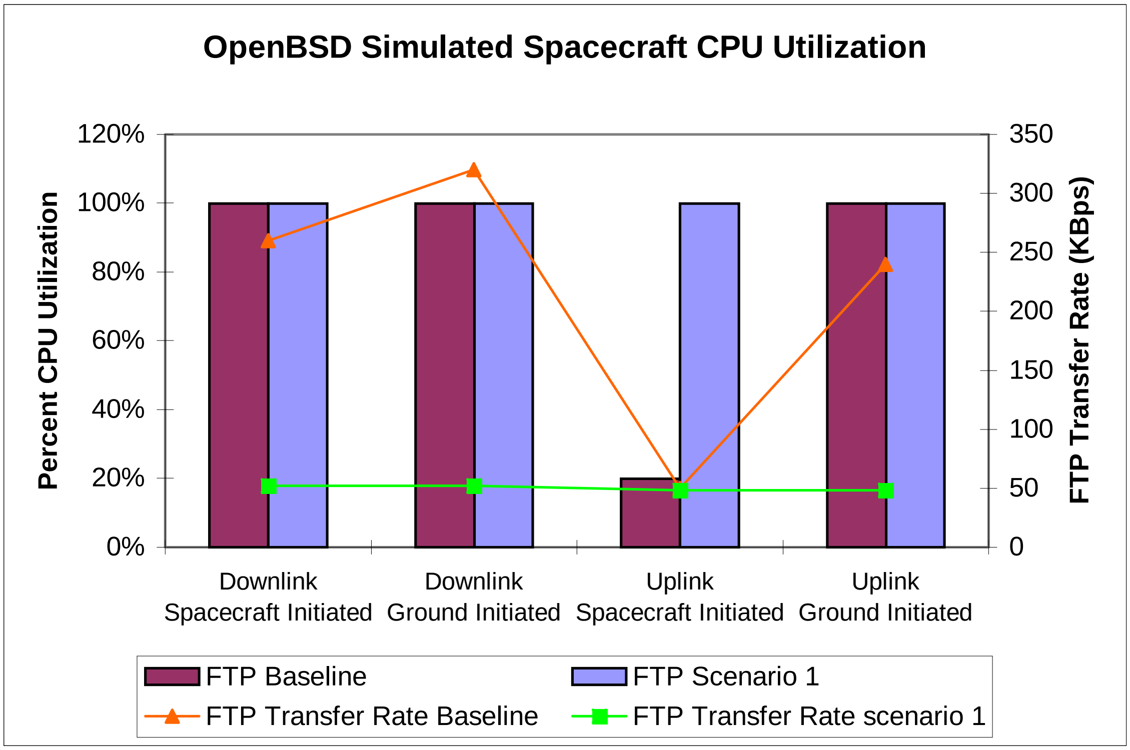


Figure 3-1. OpenBSD IPSec CPU Utilization

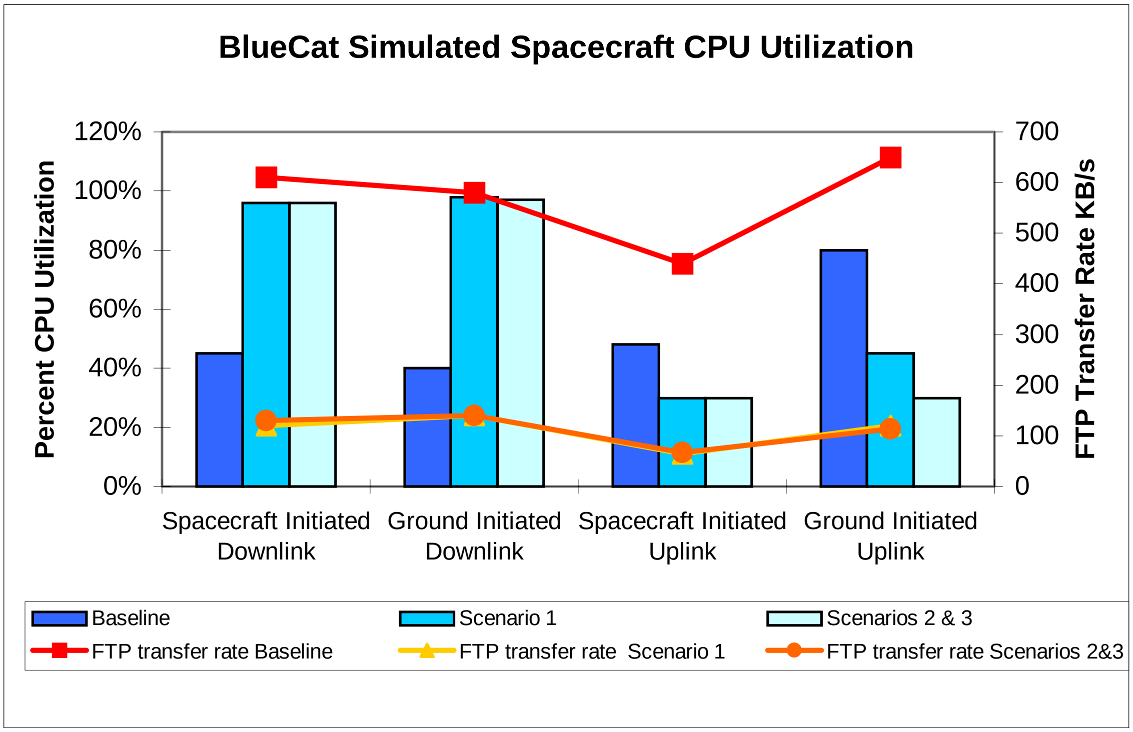


Figure 3-2. BlueCat IPSec CPU Utilization

3.2.2 Latency or Delay

Delay is added by the initial stages of IPSec, when the VPN tunnel is established. On the BlueCat system, IPSec set up takes 15 seconds plus 3 seconds thereafter for each IPSec tunnel instantiated. Once the tunnel is established, the only added delay is due to less effective throughput. The SecurID authentication process also takes time. Some delay is caused by the round trip time of packets between the user authenticating and the resource he or she authenticates to and of packets between the resource and the SecurID server. Additional delay is caused by the extra step of a human entering the password. This delay may or may not detract from time spent communicating with the spacecraft, depending on whether or not the spacecraft was the resource to which a user was authenticating.

Latency is added when encryption is used. Round trip time measured by a ping increased an average 6.2 ms when IPSec with Triple Data Encryption Standard (3DES) encryption algorithm was used. However, the latency was increased only by an average 5.3 ms when IPSec with DES or Blowfish encryption was used. The speed (and efficiency) of the encryption algorithm used does affect the bandwidth overhead and throughput, and therefore latency experienced by encrypted packets.

3.2.3 Bandwidth Utilization and Overhead

Several of the technologies in the proposed solutions add bandwidth overhead and restrict bandwidth utilization, or throughput. IPSec adds overhead to every packet. Firewalls may become network bottlenecks. SecurID authentication and other authentication mechanisms add set-up steps and possibly extra round trip communication with the spacecraft. Of these, IPSec is the only technology that significantly impacts bandwidth utilization and overhead throughout the entire course of the communication. The firewalls were able to run without slowing the traffic significantly. The SecurID and IPSec initialization steps occur only at the start of a communication, and would likely occur only at the start of a satellite pass.

3.2.3.2 IPSec Overhead

IPSec overhead varies depending on the protocol, mode, hash, and encryption options chosen. Additional headers and encryption padding make up the IPSec overhead. Tunnel mode adds 20 Bytes per packet for the second IP header it attaches to each packet. Encapsulation Security Payload (ESP) protocol headers may add 26 or 30 Bytes per packet. The Authentication Header (AH) protocol headers may add 24, 28, or 32 Bytes per packet. Encryption algorithms operate on fixed-size data blocks. The algorithm will add padding to the data to get an integer number of correctly sized blocks. IPSec will add at least 24 bytes per packet in transport mode and at least 44 bytes per packet in tunnel mode. The maximum overhead for tunnel mode AH is 52 bytes plus any IP options that are added to the new IP header. The maximum overhead for tunnel mode ESP is 60 bytes plus IP options plus encryption padding plus alignment padding plus an Initialization Vector if the encryption algorithm requires one. This overhead may double the size of tiny packets. The overhead will result in fragmentation of packets larger than the MTU minus the overhead. Wisely chosen packet sizes can help minimize fragmentation and padding necessary for encryption.

3.2.3.3 Throughput Degradation from Encryption

Throughput may dramatically decrease when encryption is used. The following table consolidates results obtained with FTP file transfers without encryption and within an IPSec ESP tunnel using 3DES encryption algorithm. The average throughput degradation is 74% for cases with a 486 PC running Linux as the simulated spacecraft.

Table 3-1. Throughput Impact of IPSec

			Transfer Rate (KBps)			Throughput Degradation	
Simulated Spacecraft Host	**Direction**	**FTP Initiator**	**Baseline**	**Concept 1**	**Concepts 2& 3**	**Concept 1**	**Concepts 2& 3**
BlueCat	Downlink	Spacecraft	610	120	130	80%	79%
BlueCat	Downlink	Ground	580	140	140	76%	76%
BlueCat	Uplink	Spacecraft	440	64	67	85%	85%
BlueCat	Uplink	Ground	650	120	114	82%	82%
OpenBSD	Uplink	Ground	240	49	n/a	80%	n/a
OpenBSD	Downlink	Ground	320	55	n/a	83%	n/a
OpenBSD	Uplink	Spacecraft	50	49	n/a	3%	n/a
OpenBSD	Downlink	Spacecraft	260	55	n/a	79%	n/a
Windows	Symmetric	N/A	924.91	739.64	n/a	20%	n/a

Some anomalous results in this table are noteworthy. The symmetric test of two Windows hosts communicating over an IPSec tunnel results in only 20% degradation. The faster Pentium III processors on the Windows hosts were better able to keep up with the demands of encryption.

In the case of the OpenBSD uplink initiated from the OpenBSD simulated spacecraft, the additional encryption did not result in a major throughput degradation. However, the baseline throughput was extremely low. In this case, the simulated spacecraft was able to encrypt the attempted communication without impacting the expected throughput by using 100% of its CPU. In the unencrypted test, it only used 20% of its CPU. Therefore, the throughput was largely unaffected, but at the expense of all the CPU resources. This affect on throughput will be greater for spacecraft with large data rate requirements, and will be less, if at all, for spacecraft with small data rate requirements.

Different encryption algorithms affect the impact of encryption on throughput. SCP from OpenSSH was used in an automated test to capture the throughput corresponding to different encryption algorithms. SSH was used because it provided the option of using Advanced Encryption Standard (AES) to encrypt. AES was not supported in the IPSec implementations tested. AES is the newest encryption standard; it is thought to be more secure and more efficient than the previous standardized algorithms DES and 3DES.

Table 3-2. Throughput Impact of Encryption Algorithms

Encryption Algorithm	**Downlink Bytes/second Transmitted**	**Uplink Bytes/second Transmitted**
aes128-cbc	83,000 to 128,000	14,000 to 18,000
aes192-cbc	84,000 to 112,000	14,000 to 18,000
aes256-cbc	83,000 to 97,000	14,000 to 17,000
blowfish	99,000 to 152,000	14,000 to 18,000
cast128-cbc	102,000 to 146,000	14,000 to 17,000
arcfour	161,000 to 178,000	15,000 to 17,000
3des	18,000 to 19,000	18,000
3des-cbc	15,000 to 16,000	10,000 to 12,000

The uplink values in Table 3-2 do not appear to vary with the encryption algorithm. This is probably due to limiting PC architecture issues. When the systems were not at maximum capacity, on the downlink trials, the difference in throughput based on the encryption algorithm is more evident. 3DES and 3DES-CBC have the worst performance; however, 3DES is supported and used in most IPSec products. It was used in the majority of IPSec tests in this prototype effort.

3.2.4 Ease of Use and Interoperability

Generally, various vendor products can be made to interact, but not always with the strongest security options available. Often the overall security is subject to the highest common level of encryption algorithms, protocols, and capabilities supported by all the tools. If one of the products used to secure the communications can support only weak encryption, the whole system suffers from that weakness. In addition, system complexity increases the chance of system failures, and increases the chance of security flaws. Whenever possible, a simple solution should be sought, and layers of security should be employed so that if a weakness in one layer is found and exploited, a secondary layer can prevent a security breach.

3.2.5 Features of Security

The IPSec standard lays out a framework for providing authentication, data integrity, and data confidentiality. However, the implementation and configuration of IPSec must be scrutinized to be sure it is using the framework appropriately to actually provide those protections. For instance, the FreeS/WAN implementation of IPSec will negotiate the least secure security association offered by a host attempting to connect. In our tests, we simply configured the clients for the appropriate encryption and hashing algorithms. However, in reality, we would not want a spacecraft to accept any security association offered. We would want it to reject all but the level we deem appropriate. Also, the Cisco clients we used stored the IPSec keys – the user never had to enter them. This effectively authenticated the host and allowed anyone using that host to access the VPN. This too may be undesirable in real scenario. In our prototype, the SecurID provided user authentication at the firewall after the IPSec tunnel was set up. Therefore, host and user authentication was accomplished. These examples show that missions must be careful to scrutinize the details of their configurations to understand exactly what protection they are or are not getting. Without such close scrutiny, the mission could gain a false sense of security and be vulnerable to attack.

NASA requires that cryptographic products be certified or endorsed by National Institute of Standards and Technology (NIST)[2] if they are included in a NASA system. Some of the specific products we tested have not yet been certified or endorsed by NIST.

Digital Signature Algorithm (DSA), DES, 3DES, and Secure Hash Algorithm (SHA) are certified by NIST as Federal Information Processing Standard (FIPS) compliant. Final NIST approval of AES may come later this year.

Testing performed by NIST accredited Cryptographic Module Testing (CMT) laboratories have validated Netscreen 100 encryption software and hardware, Cisco encryption software, and Microsoft Windows 2000 encryption software as compliant with the 3DES standard. In addition, the Netscreen 5 firewall hardware was validated as compliant with FIPS 140-1.

[2] NPG 2810.1 *Security of Information Technology, 1999.* Section 4.11.2.

Section 4. Conclusion

IP standards-based COTS security products can be used together to create communication systems with added protection against risk. We found that creating an integrated solution with many products can be difficult and complex. A lot of tweaking had to take place to make the communication work. The security adds complexity, especially when it is not an end-to-end solution, but is done piecemeal throughout the communications path. To simplify the security solution and improve system efficiency, the security should be designed into the system rather than applied after the fact.

Our proposed solutions either could not meet or had difficulty meeting the recommended protection requirements for communication paths outside redundant mission networks. Outside the mission network domains, availability is unreliable and sufficient protection of the Primary Investigator (PI) or Flight Operations Team (FOT) host computer is questionable. NASA will have to weigh the risks involved to determine if a host on an open network should have commanding access to the spacecraft or spacecraft instrument, and if so, to determine how to protect that host. Currently a Trust Model group, consisting of IT security engineers from all NASA centers, is working to create standard trust levels. Some of their standards may become useful in setting guidelines for how well a host should be protected in order to be considered trusted enough to communicate with a spacecraft.

Encryption does affect CPU resources and add significant overhead. The amount of processor utilization and packet overhead varies depending on the encryption algorithm, the IPSec mode, and the size of the unencrypted packets. Each mission will have to design their systems with these security mechanisms in mind in order to develop the most efficient set of packet size, encryption algorithm, and IPSec mode for their applications. Hardware encryption on-board the spacecraft should be explored. Field Programmable Gate Arrays (FPGA) may be useful, because they could be programmed with an encryption algorithm and keys, and they could process that algorithm in hardware. Since FPGAs are programmable, they also may provide flexibility to change algorithms or keys after a spacecraft launch.

Bandwidth is reduced when encryption is employed. Missions will have to consider their data rate requirements and plan their processor and memory resources accordingly, so that they have enough resources to encrypt and decrypt data at the rate they require.

4.1 Future Research

These results represent only the beginning of continual studies to determine what security precautions will be necessary to expand the Internet to space. We encourage the NASA community to continue pursue efforts that will explore these technologies in mission scenarios further. Currently several different groups at NASA are exploring IP in Space technologies. These groups should be encouraged to consider and study security as they design their solutions. The NASA centers should share their knowledge and studies of IP-in-Space security through collaborative efforts like the Space Internet Workshop, and similar briefings and conferences.

The concepts tested should provide the recommended security for commanding spacecraft using IP only if the integrity of the commanding hosts and availability of the open networks can be ensured. Ensuring adequate protection of a host outside NASA's purview is unfortunately an extremely difficult problem. Any host residing on an open network, such as a university LAN or the Internet is susceptible to a myriad of attacks and has a high risk of being compromised. . An IPSec tunnel may protect the communications between a host and a spacecraft. However, that IPSec tunnel can also transport IT attacks. Therefore the integrity of the hosts is critical. New IT security threats emerge daily. NASA has no way of ensuring that on a daily basis, these hosts are being patched and monitored and protected from harm. This issue must be resolved before spacecraft command is allowed to traverse open networks.

Keys are essential to encryption and authentication. If a key or a certificate is lost or compromised, a system must be in place to establish a new key. This can be extremely difficult when physical access to the spacecraft is restricted. A key cannot be sent to the spacecraft in the clear or using compromised encryption. If a list of many keys is launched with the spacecraft, the spacecraft can be commanded to simply switch to the next key on the list. If the whole list is compromised, however, there is no recovery. Certificates may be a solution, but certificates can be forged. The entire area of key management was not addressed in this study and needs careful attention.

Advances in cell phone technology have led to some low cost RF spread spectrum schemes that spread and scramble a signal across frequencies, making it hard to sniff or interfere with the wireless communication. The Advanced Range Technology Initiative (ARTI) project at WFF is currently pursuing some studies in this area. Such technologies may be another solution for providing confidentiality over the space to ground link.

Appendix A. Test Results Details

A.1 Hardware Encryption NIC cards

The IP in Space team has identified the IP Security Protocol (IPSEC) as a protocol that can supply authentication and privacy for end user data. However, the encryption process required for IPSEC is very CPU intensive. Other processing may be impacted on a host running IPSEC.

Network Interface Cards (NICs) are now available that have onboard encryption engines that can perform encryption for the host operating system. The following tests use Microsoft's Windows 2000 as the host operating system for NICs with onboard encryption engines.

One NIC is the 3Com Etherlink 10/100 Mbps PCI NIC with 3XP Processor (3CR990-TX-97) and the other NIC the Intel® PRO/100 S Server NIC. These NICs were placed in the workstations along with a standard NIC. The appropriate NIC was enabled/disabled for the test being run using the Windows 2000 network subsystem interface.

A.1.1 Test Configuration

Figure A-1 shows the configuration for the tests. The ASUS-based workstation ran the Windows 2000 Network Monitor to sample the traffic to test for encryption and proper IPSEC format.

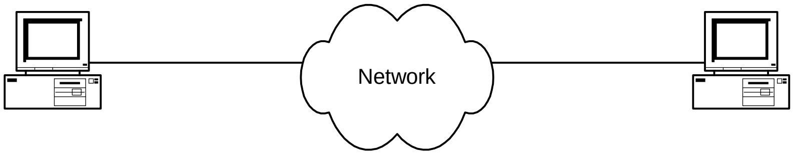


Figure A-1. Encryption NIC Test Configuration

A.1.1.1 Hardware

The tests used two networked workstations running Microsoft Windows 2000, two standard NICs for establishing a baseline and two NICS with an onboard encryption engine.

One workstation is built on a TYAN S1837UANG Thunderbolt dual processor motherboard populated with two 550MHz Intel Pentium III CPUs and 512MB of memory. The TYAN motherboard also has an onboard network connection capability using the Intel 82559 10/100 Ethernet chip. This connection was enabled, and the encryption-capable NIC disabled, for the tests requiring no encryption engine assistance. Windows 2000 Professional with Service Pack 2 is installed in the standard configuration.

The other workstation is built on an ASUS P5AWOA single processor motherboard with an Advanced Micro Devices (AMD) K6-200 MHz processor and 128MB of memory. This workstation has a Linksys EtherFast 100Base-TX LAN card installed for testing with no encryption. Windows 2000 Server with Service Pack 2 is installed in the standard configuration.

The network is a 100 Base-T full-duplex-switched network. Microsoft's Network Monitor was run on the ASUS Windows 2000 Server. Each test was run twice: once with the monitoring utilities running with the file transfer and again with only the file transfer.

No optimization was done on either system. The standard configuration settings, e.g., networking window size, were used as set by the Windows 2000 operating system installation.

A.1.1.2 Software

The standard performance measuring tools and FTP service of Microsoft Windows 2000 were used. In addition, the "ipsecmon" utility of Windows 2000 was used to obtain the IPSEC statistics for both Workstations. The Windows 2000 Service Pack 2 update file, w2ksp2.exe, was the file transferred between the workstations with FTP.

A.1.2 CPU Utilization

The following table summarizes the results of testing. The overall conclusion is that a NIC with an on-board encryption engine can off-load substantial processing from the host system if IPSEC ESP is being used. In addition, the transfer rates for the FTP transfers varied considerably for identical test runs. Therefore, these transfer rates are not a reliable measure of the effects of encryption on bandwidth using these NICs.

Table A-1. Encryption NIC CPU Utilization Benefit

Test	ASUS CPU	TYAN CPUs	Transfer Rate
Baseline – regular FTP	8.7 %	19.6 %	924.91 KBps
IPSEC ESP – no NIC engine	52.8 %	23.2 %	739.64 KBps
IPSEC ESP – with NIC engine	10.5 %	13.8 %	890.75 KBps
IPSEC AH – no NIC engine	18.2 %	17.8 %	972.94 KBps
IPSEC AH – with NIC engine	10.5 %	13.9 %	896.74 KBps

A.1.3 Ease of Use and Interoperability

The 3Com NIC may be used with Linux, SCO Openserver, NetWare, SCO UnixWare, and Windows systems. The Intel NIC may be used with Windows, NetWare, Solaris, SCO UnixWare, and Linux systems.

During initial testing, it was discovered that encryption processing was not being off-loaded to the 3com Etherlink 10/100 Mbps PCI NIC with 3XP Processor (3CR990-TX-97) using EL98XND5.SYS version 1.01.21.0000 when installed in the dual processor TYAN-based system. The 3com NIC was swapped with the Intel® PRO/100 S Server NIC using iansw2k.sys version 01.51.00.000, ivlanw2k.sys version 2.27.00.000, prod.sys version 1.15, and prokddp.sys version 2.21 in the ASUS-based system. The Intel NIC did perform encryption processing on the dual processor TYAN-based system. The 3Com NIC did perform encryption processing on the ASUS-based system.

No further investigation was conducted with the 3Com NIC on the TYAN–based system other than downloading and installing the latest drivers from 3Com. The problem remained using the newest drivers.

A.1.4 Features of Security

The NICs do not provide security in and of themselves. The security is provided by the protocols or technologies that are making use of the encryption process. In this case, IPSec was used to encrypt traffic. Comments on the degree of security of IPSec are made in following sections.

Both encryption NICs support 3DES and DES encryption algorithms and MD5 and SHA-1 hashing algorithms. DES is considered an outdated algorithm too easily broken. 3DES was the standard best-practice algorithm until AES was chosen as its successor. Neither card support AES. The NICs are not NIST FIPS certified.

A.1.5 Test 1 – No Encryption Baseline

A.1.5.1 Software Setup

Windows 2000 was configured to establish an unencrypted link between the two workstations. This was done for both workstations by enabling the ordinary NIC, disabling the NIC with the encryption engine, and then selecting the IP security policy to allow unencrypted traffic.

NICs may be enabled or disabled by right-clicking “My Network Places”, selecting “Properties” from the pop-up menu, right-clicking the icon for the desired NIC, and then selecting “Enable” or “Disable” from the pop-up menu. Select unencrypted or encrypted communications by opening the “Local Security Policy” Administrative Tool, selecting “IP Security Policies” in the left-pane, right-clicking either “Client (Respond Only)” or “Lockdown” in the right-pane for unencrypted or encrypted communications, respectively, and selecting “Assign” from the pop-up menu. The “IP Policy Agent” must be restarted using the “Services” control panel to ensure that the policy change is read.

A.1.5.2 Testing Procedure

1. Synchronize the clocks on the two workstations by issuing the following command in a command prompt window on the workstation with the secondary clock:
    a. Net time \\<other-workstation-name> /set
        i. To find the <other-workstation-name>, right-click “My Computer” on the other workstation and select the “Network Identification” tab. The name is after “Full computer name”.
    b. Respond “y” to the confirmation message
2. Activate the Microsoft Network Monitor on the ASUS-based workstation to capture frames between the two workstations.
3. Open the Performance control panel in the “Administrative Tools” on both workstations and,
    a. Double-click “Performance Logs and Alerts” and then click “Counter Logs”
    b. Right-click a blank area of the details pane and click “New Log Settings”
    c. In “Name” type the name of the log and then click “OK”
    d. On the “General” tab select the “% Processor Time” counter and add each of the CPUs by selecting it in the right pane and clicking “Add”
    e. Click “Close”
    f. Adjust the Interval on the “General” tab to 5 seconds and click “Apply”. If a prompt opens asking whether it is OK to create the directory, click “Yes”
    g. Click “OK”
4. Activate ipsecmon on both workstations. Do a “Print Screen” holding down the “Alt” key to get only the active window and save it as a bitmap to a file.
5. Start the “Performance Monitors” on both workstations.

6. Open a command window on the TYAN-based workstation and type “ftp <other-workstation-IP>”, submit a valid username/password and type “binary”. Issue the command “get W2ksp2.exe” to transfer the file from the ASUS-based workstation. The default directory on the ASUS-based workstation is whatever it has been set to in the “Internet Information Server” settings in the “Computer Management” control panel.
7. Wait for the transfer to complete.
8. Do a “Print Screen” of the ftp session command window to a file on the TYAN-based workstation.
9. Stop the Performance Monitor on both workstations and save the latest log file on each workstation.
10. Do a “Print Screen”, holding down the “Alt” key to get only the active window, of the “Ipsecmon” screen to a file.
11. Open the “w2ksp2.exe” file by double clicking to verify it is intact. The file will check its integrity and present a message. Abort the installation of the service pack.
12. Repeat the test without activating the Microsoft Network Monitor or the Ipsecmon utility. Then do a “Print Screen” on the TYAN-based workstation of the FTP transfer window and save the “Performance Monitor” log files on both workstations.

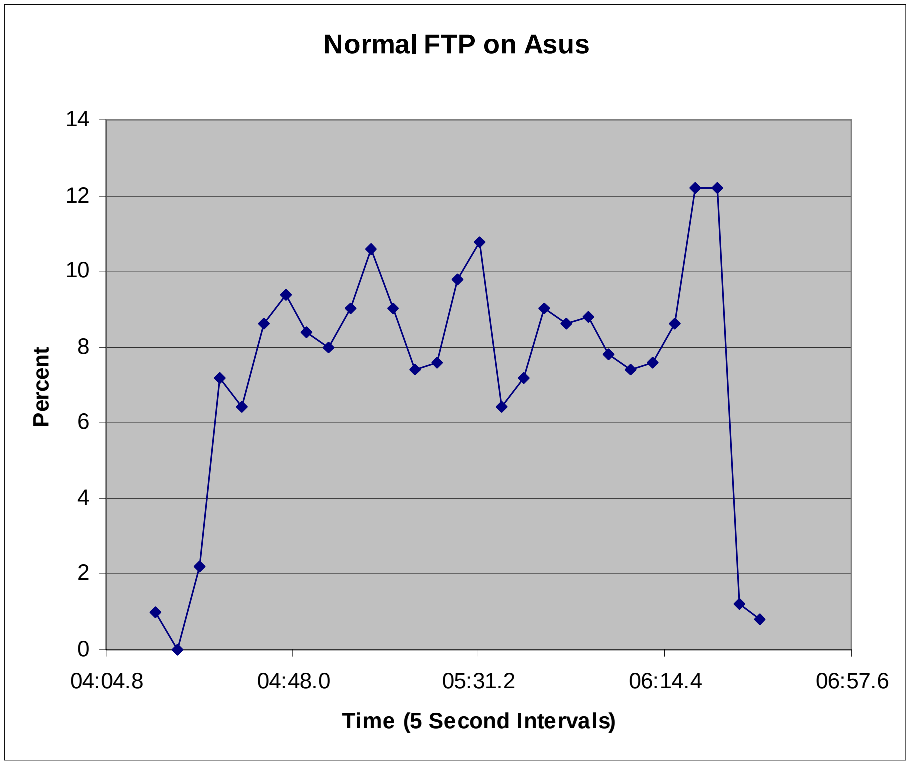
A.1.5.3 Expected Results

- The file transfer rate in kilobytes per second will be known.
- The CPU usage on the workstations will be known.
- The Microsoft Network Monitor output will verify that the frames are properly formatted and the data is in the clear.
- Any anomalies will be documented.

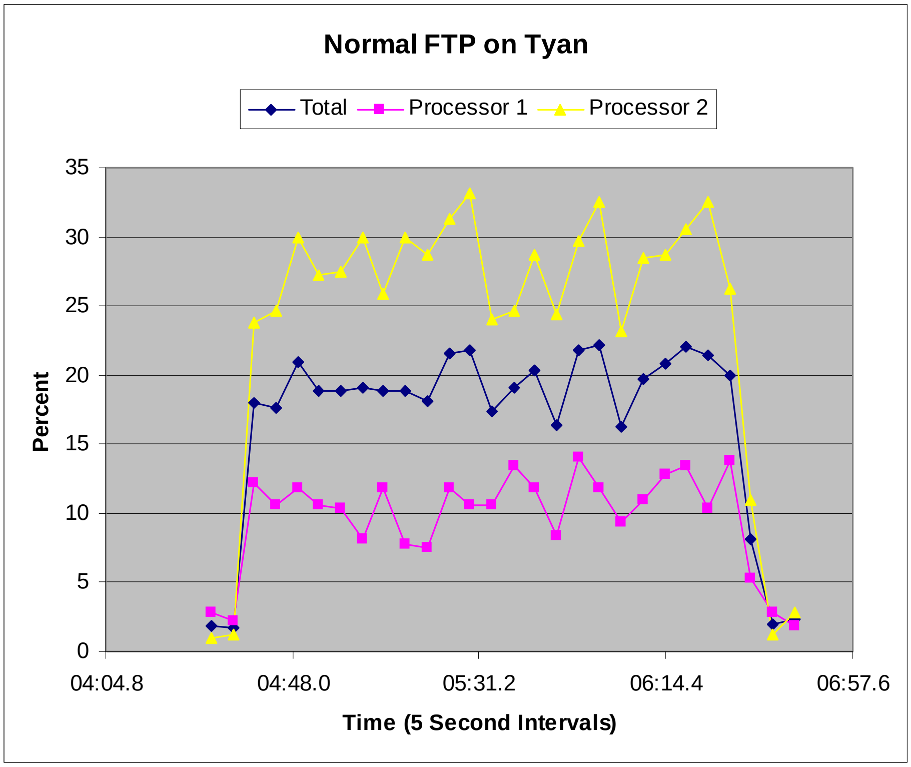
A.1.5.4 Results

The ftp transfer was 106,278,016 bytes transferred in 114.91 seconds, for a rate of 924.91 kilobytes per second.

The ASUS had the following CPU usage:



The TYAN had the following CPU usage:



The total CPU usage of the TYAN-based system is calculated by summing the time both processors are used and then dividing by the total time available for use. It effectively averages the CPU utilization since both processors have the same clock-rate.

As expected, Ipsecmon, the Windows 2000 Security Association monitor, showed no activity for IPSEC.

The Network Monitor, part of Windows 2000 Server, captured 1,100 packets for the session. No artifacts of the IPSEC protocol appeared in the captured frames. The following is a sample frame from the captured frames. It is clearly identifiable as an FTP session. The contents of the file being transferred are not readable because the file is a binary file.

```
1 319.249057 0050DA6E21B7 0002FD14DF40 FTP Data Transfer To Client, Port = 1122, size
1456 192.168.200.2 10.0.0.14 IP

Frame: Base frame properties
    Frame: Time of capture = 5/23/2001 18:59:58.160
    Frame: Time delta from previous physical frame: 0 microseconds
    Frame: Frame number: 1
    Frame: Total frame length: 1510 bytes
    Frame: Capture frame length: 1510 bytes
    Frame: Frame data: Number of data bytes remaining = 1510 (0x05E6)
ETHERNET: ETYPE = 0x0800 : Protocol = IP:  DOD Internet Protocol
    ETHERNET: Destination address : 0002FD14DF40
        ETHERNET: .......0 = Individual address
        ETHERNET: ......0. = Universally administered address
    ETHERNET: Source address : 0050DA6E21B7
        ETHERNET: .......0 = No routing information present
        ETHERNET: ......0. = Universally administered address
    ETHERNET: Frame Length : 1510 (0x05E6)
    ETHERNET: Ethernet Type : 0x0800 (IP:  DOD Internet Protocol)
    ETHERNET: Ethernet Data: Number of data bytes remaining = 1496 (0x05D8)
IP: ID = 0xCA84; Proto = TCP; Len: 1496
    IP: Version = 4 (0x4)
    IP: Header Length = 20 (0x14)
    IP: Precedence = Routine
    IP: Type of Service = Normal Service
    IP: Total Length = 1496 (0x5D8)
    IP: Identification = 51844 (0xCA84)    IP: Flags Summary = 2 (0x2)
        IP: .......0 = Last fragment in datagram
        IP: ......1. = Cannot fragment datagram
    IP: Fragment Offset = 0 (0x0) bytes
    IP: Time to Live = 128 (0x80)
    IP: Protocol = TCP - Transmission Control
    IP: Checksum = 0x97E2
    IP: Source Address = 192.168.200.2
    IP: Destination Address = 10.0.0.14
    IP: Data: Number of data bytes remaining = 1476 (0x05C4)
TCP:  .AP...,  len:  1456,  seq:4011456583-4011458039,  ack:2165597024,  win:17472,  src:
20  dst: 1122
    TCP: Source Port = FTP [default data]
    TCP: Destination Port = 0x0462
    TCP: Sequence Number = 4011456583 (0xEF19F847)
    TCP: Acknowledgement Number = 2165597024 (0x81146360)
    TCP: Data Offset = 20 (0x14)
    TCP: Reserved = 0 (0x0000)
    TCP: Flags = 0x18 : .AP...
        TCP: ..0..... = No urgent data
        TCP: ...1.... = Acknowledgement field significant
        TCP: ....1... = Push function
        TCP: .....0.. = No Reset
        TCP: ......0. = No Synchronize
        TCP: .......0 = No Fin
    TCP: Window = 17472 (0x4440)
    TCP: Checksum = 0x8163
    TCP: Urgent Pointer = 0 (0x0)
    TCP: Data: Number of data bytes remaining = 1456 (0x05B0)
FTP: Data Transfer To Client, Port = 1122, size 1456
    FTP: FTP Data: Number of data bytes remaining = 1456 (0x05B0)
00000:  00 02 FD 14 DF 40 00 50 DA 6E 21 B7 08 00 45 00   ..ý.ß@.PÚn!·..E.
00010:  05 D8 CA 84 40 00 80 06 97 E2 C0 A8 C8 02 0A 00   .ØÊ„@. .—âÀ¨È...
00020:  00 0E 00 14 04 62 EF 19 F8 47 81 14 63 60 50 18   .....bï.øG .c`P.
00030:  44 40 81 63 00 00 E9 38 A1 ED E3 D0 97 95 CF DD   D@ c..é8¡íãÐ—•ÏÝ
…
005A0:  A0 F5 D2 89 A7 56 D3 96 3F B2 BC 3E C0 93 6E C3    õÒ‰§VÓ–?²¼>À“nÃ
005B0:  2C FF A9 8D 26 BD C2 29 F6 5F 12 AF 88 33 D1 A5   ,ÿ© &½Â)ö_.¯ˆ3Ñ¥
005C0:  4F E1 E6 50 A6 D8 F6 EC 00 27 CD 2F 5C 5A F9 0F   OáæP¦Øöì.'Í/\Zù.
005D0:  15 FD A1 4D 93 26 59 F5 5C D8 01 4A A3 0C A4 7B   .ý¡M“&Yõ\Ø.J£.¤{
005E0:  E9 52 A8 F0 A7 23                                 éR¨ð§#
```

A.1.6 Test 2 – ESP Without NIC Encryption Engine

A.1.6.1 Software Setup

Windows 2000 was configured to establish an encrypted link between the two workstations using IPSEC ESP using the AH/ESP Procedure. This was done for both workstations by disabling the NIC with the encryption engine, enabling the ordinary NIC, and then following the AH/ESP Procedure to create and enable the security policy requiring encryption.

Disable the NIC with the encryption engine, enable the conventional NIC, and enable encrypted communications by following the procedures outlined in section A.1.5.1.

The ASUS-based workstation running the Microsoft Network Monitor captured the frames of the file transfer.

A.1.6.2 Testing Procedure

The procedure specified in section A.1.5.2 was repeated.

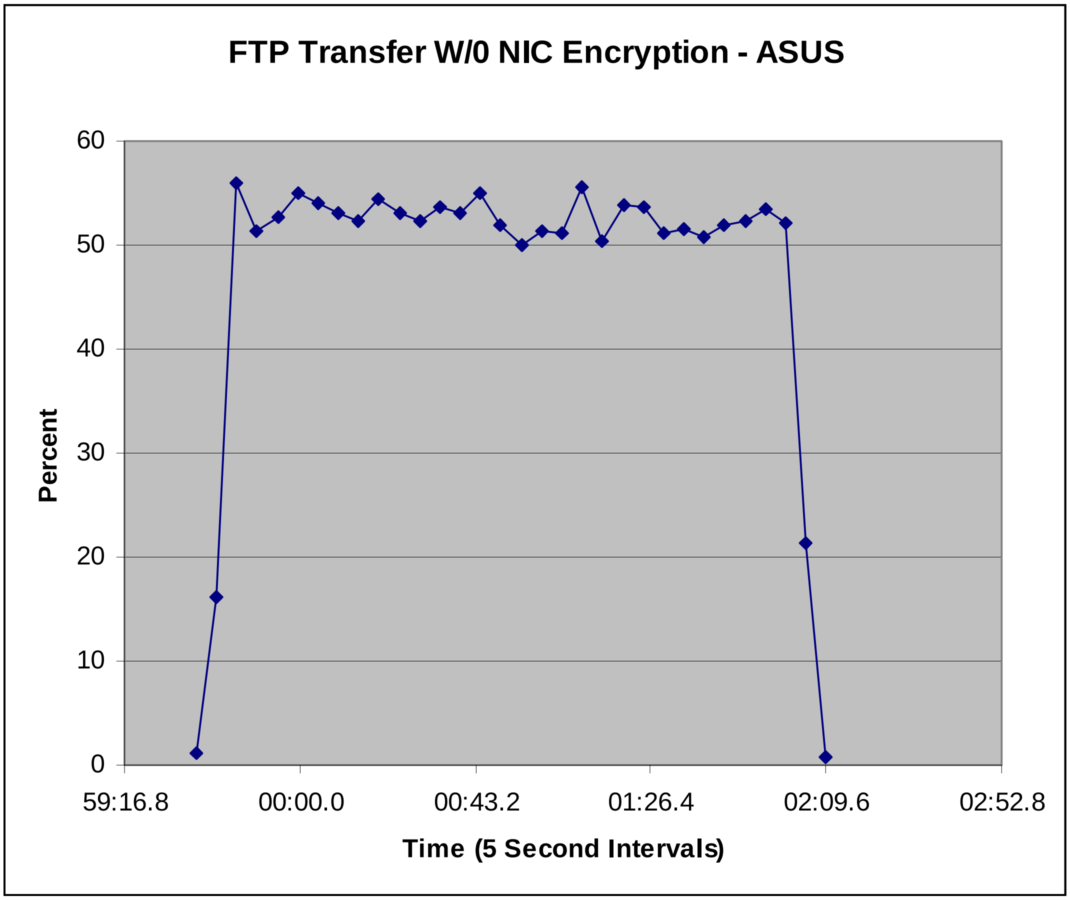
A.1.6.3 Expected Results

- The AH/ESP Procedure will be corrected.
- The file transfer rate in kilobytes per second will be known.
- The CPU usage on the workstations will be known.
- The Microsoft Network Monitor output will verify that the IPSEC frames are properly formatted and the data is not in the clear.
- The IPSEC statistics will be known from the “ipsecmon” screenshots.
- Any anomalies will be documented.

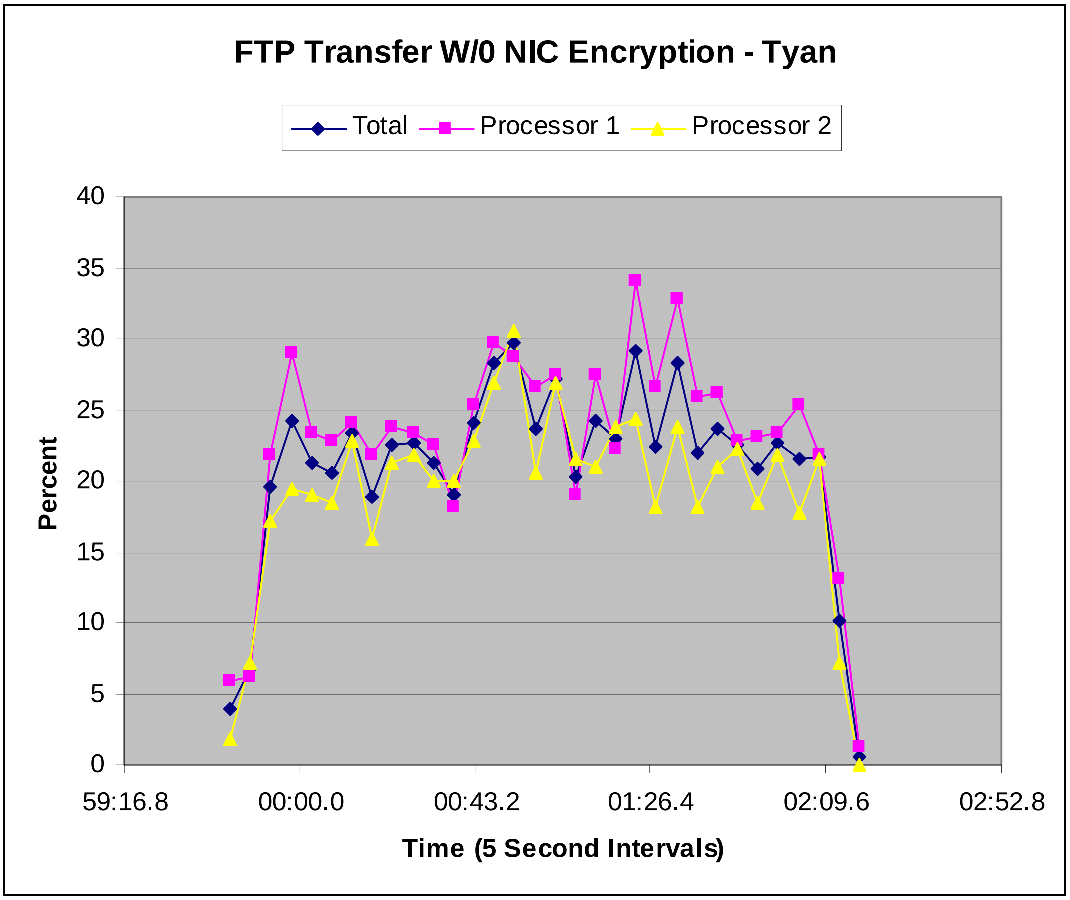
A.1.6.4 Results

The ftp transfer was for 106,278,016 bytes transferred in 143.69 seconds, for a rate of 739.64 kilobytes per second.

The ASUS had the following CPU usage:



The Performance monitor calculates a total processor usage that is the total CPU used divided by the total CPU available. The TYAN had the following CPU usage:



The total CPU usage of the TYAN-based system is calculated by adding the time both processors are used and dividing by the total time available. It effectively averages the CPU utilization since both processors have the same clock-rate.

The table shows that the process starts with the creation of the active security association. This step may or may not appear on the later tests due to the security association already having been established. The number of bytes varies probably due to additional Microsoft Windows activity.

IPSEC Statistic	**TYAN**	**ASUS**
Active Associations	1	1
Confidential Bytes Sent	746,971	107,782,663
Confidential Bytes Received	107,782,596	747,037
Authenticated Bytes Sent	1,493,168	109,361,976
Authenticated Bytes Received	109,361,848	1,493,296
Key Additions	2	2
Oakley Main Modes	1	1
Oakley Quick Modes	2	2

The Network Monitor, part of Windows 2000 Server, captured 1,000 ESP packets for the session with the following being the first packet. None of the packets were identifiable as FTP packets. This is due to the IPSec ESP protocol.

```
2 243.159647 ANI CO35D365 0002FD14DF50 ESP SPI = 0x1A8A71F9, Seq = 0x4C77 10.0.0.4
192.168.200.14 IP
Frame: Base frame properties
    Frame: Time of capture = 5/23/2001 17:51:52.114
    Frame: Time delta from previous physical frame: 0 microseconds
    Frame: Frame number: 2
    Frame: Total frame length: 1510 bytes
    Frame: Capture frame length: 1510 bytes
    Frame: Frame data: Number of data bytes remaining = 1510 (0x05E6)
ETHERNET: ETYPE = 0x0800 : Protocol = IP:  DOD Internet Protocol
    ETHERNET: Destination address : 0002FD14DF50
        ETHERNET: .......0 = Individual address
        ETHERNET: ......0. = Universally administered address
    ETHERNET: Source address : 00400535D365
        ETHERNET: .......0 = No routing information present
        ETHERNET: ......0. = Universally administered address
    ETHERNET: Frame Length : 1510 (0x05E6)
    ETHERNET: Ethernet Type : 0x0800 (IP:  DOD Internet Protocol)
    ETHERNET: Ethernet Data: Number of data bytes remaining = 1496 (0x05D8)
IP: ID = 0x2416; Proto = 0x32; Len: 1496
    IP: Version = 4 (0x4)
    IP: Header Length = 20 (0x14)
    IP: Precedence = Routine
    IP: Type of Service = Normal Service
    IP: Total Length = 1496 (0x5D8)
    IP: Identification = 9238 (0x2416)
    IP: Flags Summary = 2 (0x2)
        IP: .......0 = Last fragment in datagram
        IP: ......1. = Cannot fragment datagram
    IP: Fragment Offset = 0 (0x0) bytes
    IP: Time to Live = 128 (0x80)
    IP: Protocol = Encapsulating Security Protocol
    IP: Checksum = 0x3E23
    IP: Source Address = 10.0.0.4
    IP: Destination Address = 192.168.200.14
    IP: Data: Number of data bytes remaining = 1476 (0x05C4)
ESP: SPI = 0x1A8A71F9, Seq = 0x4C77
    ESP: Security Parameters Index = 445280761 (0x1A8A71F9)
    ESP: Sequence Number = 19575 (0x4C77)
    ESP: Rest of Frame: Number of data bytes remaining = 1468 (0x05BC)
00000:  00 02 FD 14 DF 50 00 40 05 35 D3 65 08 00 45 00   ..ý.ßP.@.5Óe..E.
00010:  05 D8 24 16 40 00 80 32 3E 23 0A 00 00 04 C0 A8   .Ø$.@. 2>#....À¨
00020:  C8 0E 1A 8A 71 F9 00 00 4C 77 F0 EA 76 1B E0 32   È..Šqù..Lwðêv.à2
00030:  FA DA D9 D0 06 F8 D7 00 C2 9F E1 BC AF 54 8E 98   úÚÙÐ.ø×.ÂŸá¼¯T ˜
00040:  A0 4B BC D4 3F 5A 03 99 5E CE 2C 55 1A 94 CD AF    K¼Ô?Z.™^Î,U.”Í¯

…

00580:  E5 07 A1 C2 18 51 3D 1A 7D B1 11 4C 41 D9 26 86   å.¡Â.Q=.}±.LAÙ&†
00590:  3D 18 50 F4 57 EF 00 71 61 66 EB 32 2F AB E7 D2   =.PôWï.qafë2/«çÒ
005A0:  D1 84 FF 7A C9 47 27 1F 9B 08 A4 E3 D8 39 B0 F5   Ñ„ÿzÉG'.›.¤ãØ9°õ
005B0:  3D 4D 4B 1B ED 35 91 EF 53 2C 85 CD 36 21 93 81   =MK.í5‘ïS,…Í6!“
005C0:  D3 0B 41 98 D2 17 42 00 B5 2B 17 58 3B B5 5A DC   Ó.A˜Ò.B.µ+.X;µZÜ
005D0:  B0 30 3A 3A C1 71 48 73 E2 A4 A3 EE 40 65 CF 9D   °0::ÁqHsâ¤£î@eÏ
005E0:  E0 6E BF F2 06 9E                                 àn¿ò.
```

A.1.7 Test 3 – ESP with NIC Encryption Engine

A.1.7.1 Hardware Setup

The 2 workstations, A and B, will be configured with the onboard encryption NICs installed and set with the appropriate network configuration.

The 3com NIC in the TYAN was switched with the Intel NIC in the ASUS due to preliminary testing showing that the TYAN did not off-load encryption processing to the 3com NIC. Switching the 2 NICs corrected the problem. The 3com NIC worked properly in the ASUS-based single processor workstation and the Intel NIC worked properly in the dual processor TYAN-based workstation.

A.1.7.2 Software Setup

Windows 2000 was configured to establish an encrypted link between the two workstations using IPSEC ESP using the AH/ESP Procedure. This was done for both workstations by enabling the NIC with the encryption engine, disabling the ordinary NIC, and then following the AH/ESP Procedure to create and enable the security policy requiring encryption.

Enable the NIC with the encryption engine, disable the conventional NIC, and enable encrypted communications by following the procedures outlined in section A.1.5.1.

The ASUS-based workstation running the Microsoft Network Monitor captured the frames of the file transfer.

A.1.7.3 Testing Procedures

The procedure specified in section A.1.5.2 was repeated.

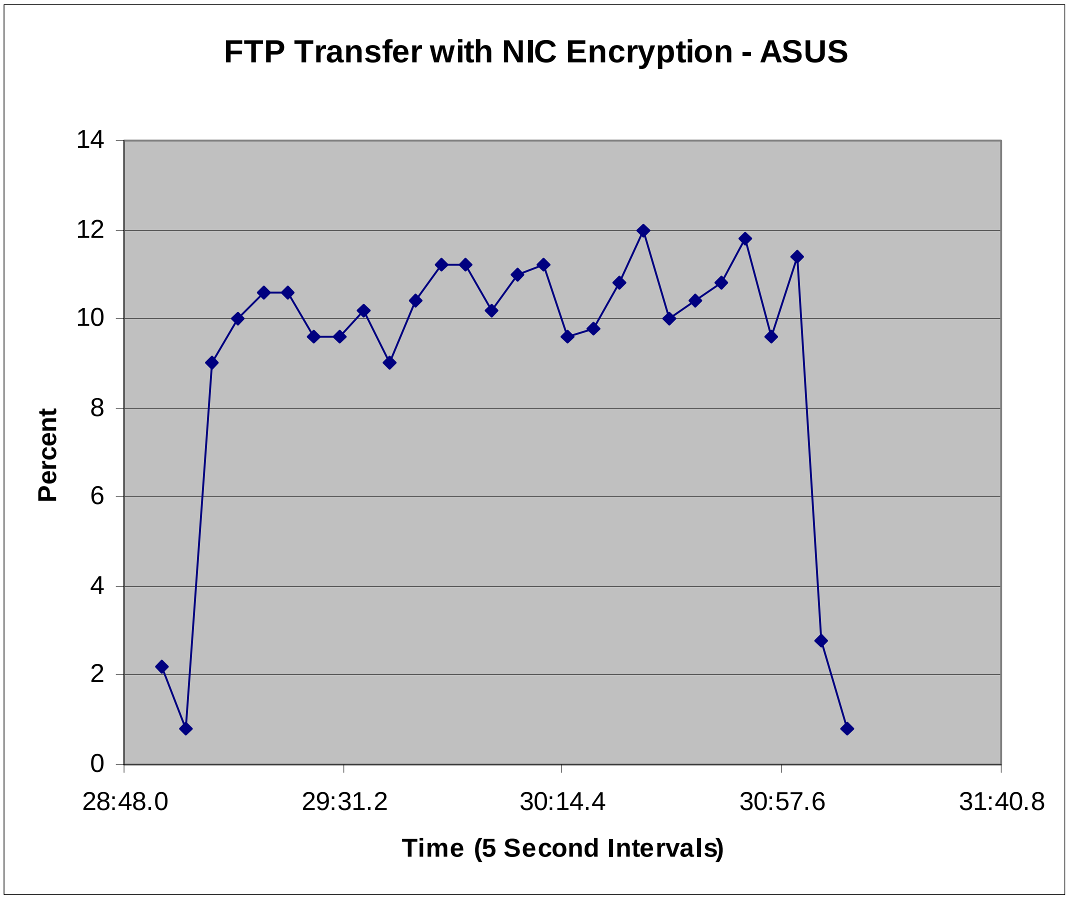
A.1.7.4 Expected Results

- The file transfer rate in kilobytes per second will be known.
- The CPU usage on the workstations will be known.
- The Microsoft Network Monitor output will verify that the IPSEC frames are properly formatted and the data is not in the clear.
- The IPSEC statistics will be known from the “ipsecmon” screenshots.
- Any anomalies will be documented.

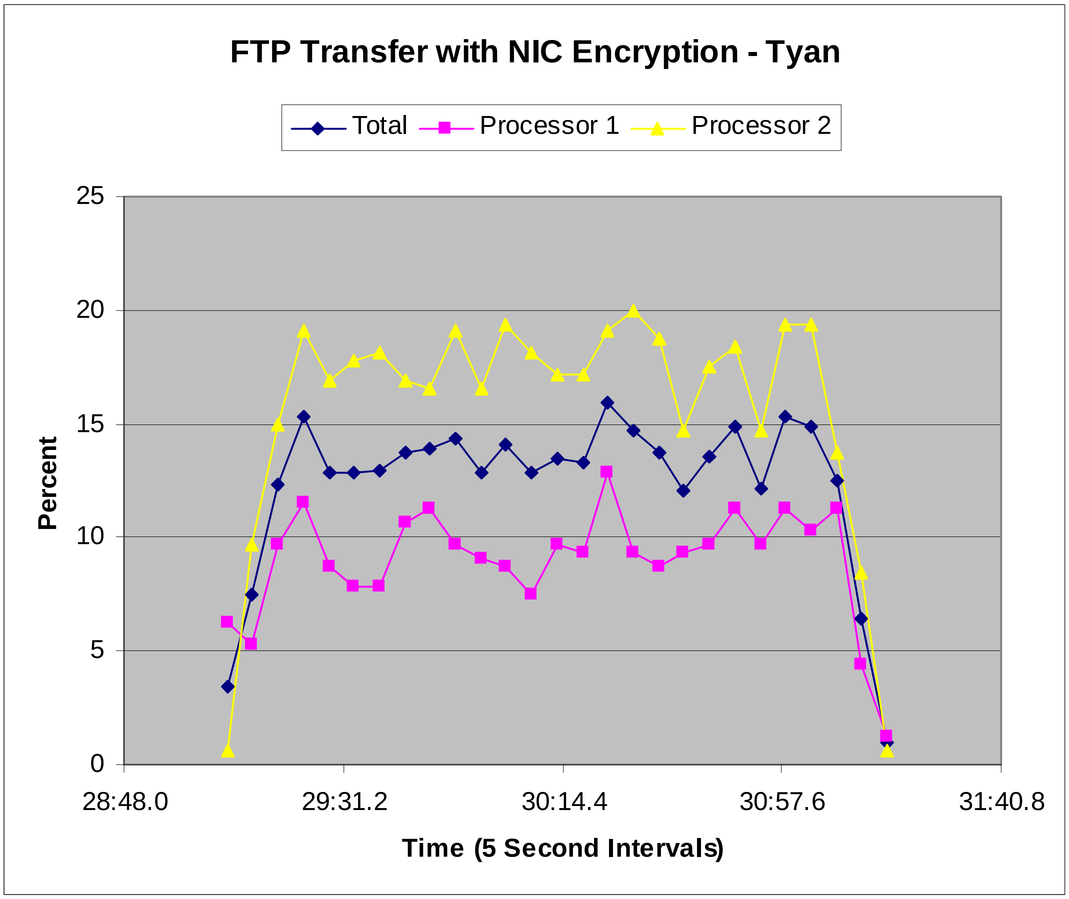
A.1.7.5 Results

The ftp transfer was for 106,278,016 bytes transferred in 119.31 seconds, for a rate of 890.75 kilobytes per second.

The ASUS CPU usage was as follows:



The TYAN CPU usage was as follows:



The total CPU usage of the TYAN-based system is calculated by adding the time both processors are used and dividing by the total time available. It effectively averages the CPU utilization since both processors have the same clock-rate.

Ipsecmon, the Windows 2000 Security Association monitor reported the following statistics:

IPSEC Statistic	TYAN	ASUS
Confidential Bytes Sent	979,494	107,783,286
Confidential Bytes Received	107,778,985	979,560
Authenticated Bytes Sent	2,742336	110,858,864
Authenticated Bytes Received	109,652,560	1,958,992
Key Additions	2	2
Oakley Quick Modes	2	2

The number of bytes varies probably due to additional Microsoft Windows activity. The Network Monitor, part of Windows 2000 Server, captured 1,000 ESP packets for the session. None of the packets were identifiable as FTP packets.

A.1.8 Test 4 – AH Without NIC Encryption Engine

A.1.8.1 Hardware Setup

The 2 workstations were configured with the standard NICs installed and set with the appropriate network configuration.

A.1.8.2 Software Setup

Windows 2000 was configured to establish an encrypted link between the two workstations using IPSEC AH using the AH/ESP Procedure. The Authentication Header option must be selected in steps 26 and 57. This was done for both workstations by enabling the ordinary NIC, disabling the NIC with the encryption engine, and then following the AH/ESP Procedure to create and enable the security policy requiring encryption.

Enable the conventional NIC, disable the NIC with the encryption engine, and enable encrypted communications by following the procedures outlined in section A.1.5.1.

The ASUS-based workstation running the Microsoft Network Monitor captured the frames of the file transfer.

A.1.8.3 Testing Procedure

The procedure specified in section A.1.5.2 was repeated.

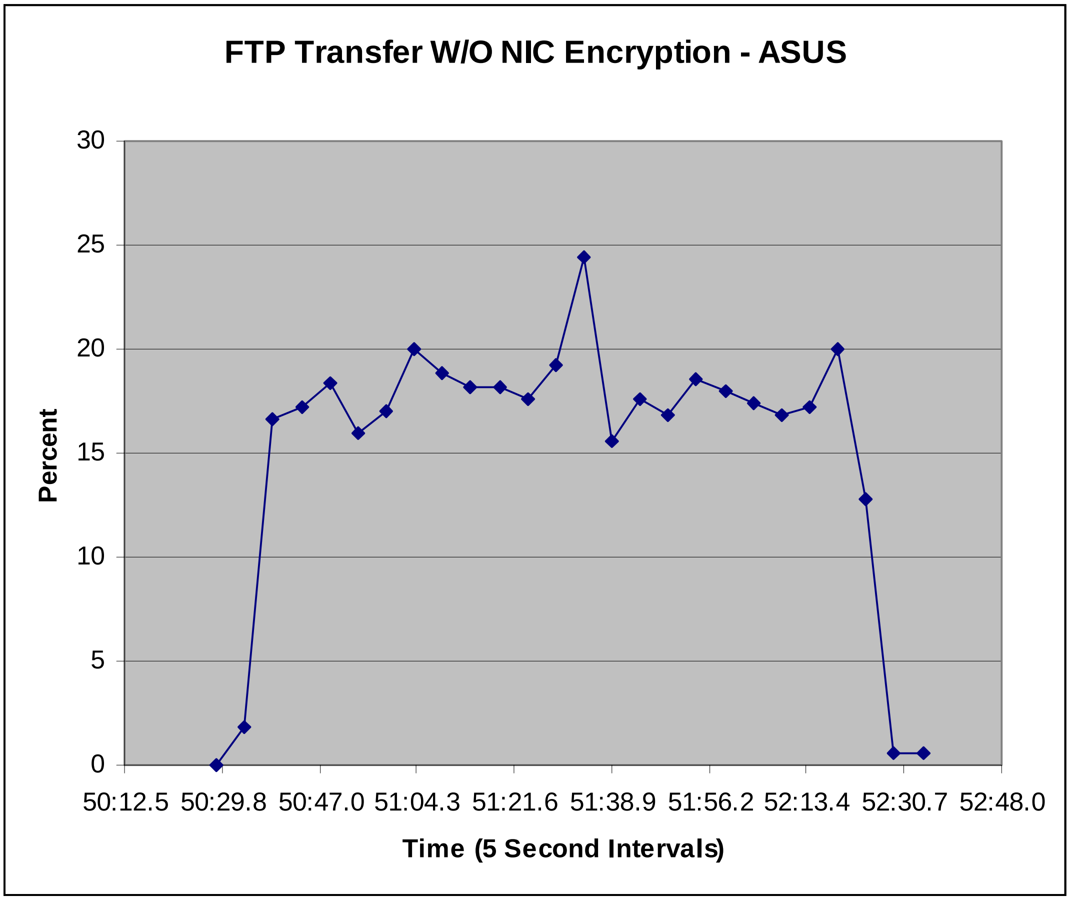
A.1.8.4 Expected Results

- The file transfer rate in kilobytes per second will be known.
- The CPU usage on the workstations will be known.
- The Microsoft Network Monitor output will verify that the IPSEC frames are properly formatted and that AH is being used.
- The IPSEC statistics will be known from the “ipsecmon” screenshots.
- Any anomalies will be documented.

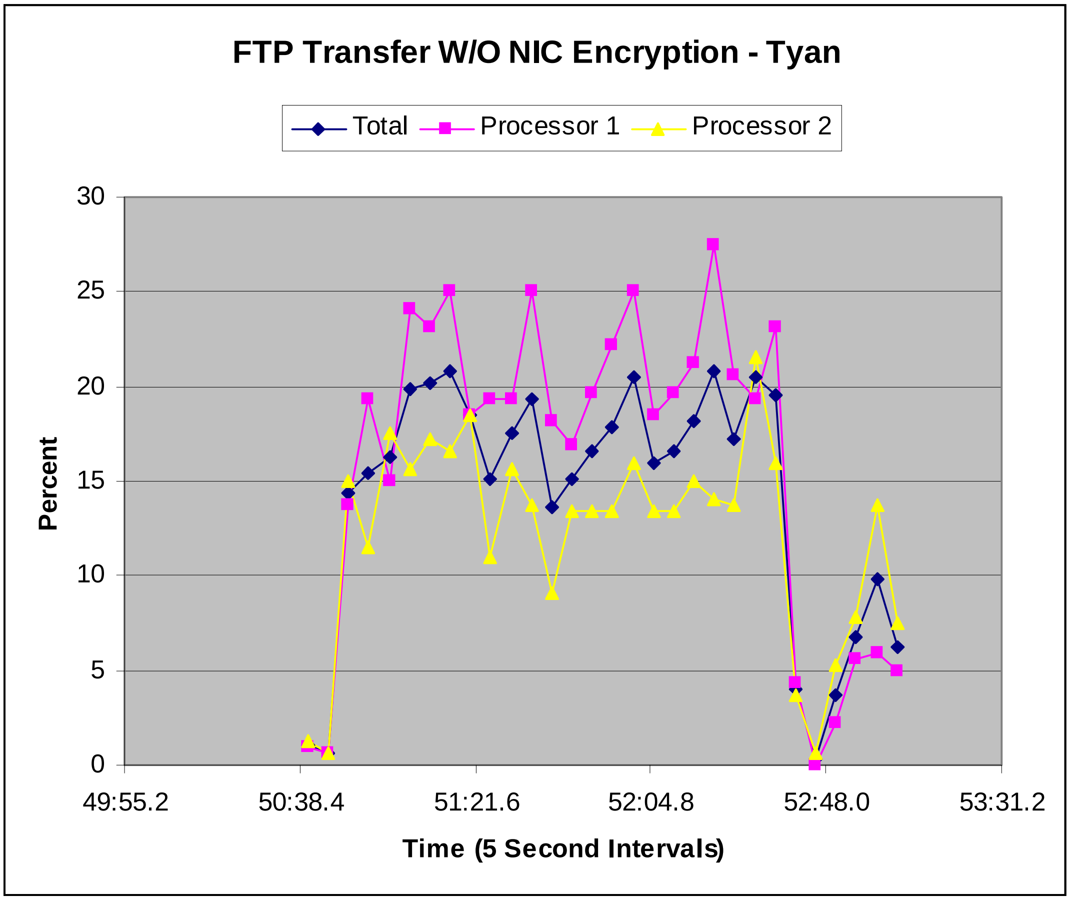
A.1.8.5 Results

The ftp transfer was for 106,278,016 bytes transferred in 109.23 seconds, for a rate of 972.94 kilobytes per second.

The ASUS CPU usage was as follows:



The TYAN CPU usage was as follows:



The total CPU usage of the TYAN-based system is calculated by adding the time both processors are used and dividing by the total time available. It effectively averages the CPU utilization since both processors have the same clock-rate.

Ipsecmon, the Windows 2000 Security Association monitor reported the following statistics:

IPSEC Statistic	TYAN	ASUS
Authenticated Bytes Sent	2,330,315	111,035,941
Authenticated Bytes Received	109,063,178	2,373,057
Key Additions	1	1
Oakley Quick Modes	1	1

The number of bytes varies probably due to additional Microsoft Windows activity. The Network Monitor, part of Windows 2000 Server, captured 1,000 AH packets for the session with the following being the first packet. It is identifiable as an FTP packet. IPSec AH protocol does not encrypt the data payload of packets. It provides authentication for the packet header.

```
1 164.286232 NTASUS 0002FD14DF50 FTP Data Transfer To Client, Port = 1080, size 1436
NTASUS 192.168.200.14 IP
Frame: Base frame properties
    Frame: Time of capture = 5/24/2001 17:43:5.302
    Frame: Time delta from previous physical frame: 0 microseconds
    Frame: Frame number: 1
    Frame: Total frame length: 1514 bytes
    Frame: Capture frame length: 1514 bytes
    Frame: Frame data: Number of data bytes remaining = 1514 (0x05EA)
ETHERNET: ETYPE = 0x0800 : Protocol = IP:  DOD Internet Protocol
    ETHERNET: Destination address : 0002FD14DF50
        ETHERNET: .......0 = Individual address
        ETHERNET: ......0. = Universally administered address
    ETHERNET: Source address : 00400535D365
        ETHERNET: .......0 = No routing information present
        ETHERNET: ......0. = Universally administered address
    ETHERNET: Frame Length : 1514 (0x05EA)
    ETHERNET: Ethernet Type : 0x0800 (IP:  DOD Internet Protocol)
    ETHERNET: Ethernet Data: Number of data bytes remaining = 1500 (0x05DC)
IP: ID = 0x7D35; Proto = 0x33; Len: 1500
    IP: Version = 4 (0x4)
    IP: Header Length = 20 (0x14)
    IP: Precedence = Routine
    IP: Type of Service = Normal Service
    IP: Total Length = 1500 (0x5DC)
    IP: Identification = 32053 (0x7D35)
    IP: Flags Summary = 2 (0x2)
        IP: .......0 = Last fragment in datagram
        IP: ......1. = Cannot fragment datagram
    IP: Fragment Offset = 0 (0x0) bytes
    IP: Time to Live = 128 (0x80)
    IP: Protocol = 0x33
    IP: Checksum = 0xE4FF
    IP: Source Address = 10.0.0.3
    IP: Destination Address = 192.168.200.14
    IP: Data: Number of data bytes remaining = 1480 (0x05C8)
AH: Protocol = TCP, SPI = 0x1EDE5AEE, Seq = 0x5D93
    AH: Next Header = TCP - Transmission Control
    AH: Payload Length = 6 (0x6)
    AH: Reserved = 0 (0x0)
    AH: Security Parameters Index = 517888750 (0x1EDE5AEE)
    AH: Sequence Number = 23955 (0x5D93)
    AH: Authentication Data: Number of data bytes remaining = 12 (0x000C)
TCP:  .A....,  len:65515,  seq:2210677049-2210742564,  ack:3983602327,  win:17520,  src:
20  dst: 1080
    TCP: Source Port = FTP [default data]
    TCP: Destination Port = 0x0438
    TCP: Sequence Number = 2210677049 (0x83C44139)
    TCP: Acknowledgement Number = 3983602327 (0xED70F297)
    TCP: Data Offset = 20 (0x14)
    TCP: Reserved = 0 (0x0000)
    TCP: Flags = 0x10 : .A....
        TCP: ..0..... = No urgent data
        TCP: ...1.... = Acknowledgement field significant
        TCP: ....0... = No Push function
        TCP: .....0.. = No Reset
        TCP: ......0. = No Synchronize
        TCP: .......0 = No Fin
    TCP: Window = 17520 (0x4470)
    TCP: Checksum = Frame was truncated, unable to verify Checksum.
    TCP: Urgent Pointer = 0 (0x0)
    TCP: Data: Number of data bytes remaining = 1436 (0x059C)
FTP: Data Transfer To Client, Port = 1080, size 1436
    FTP: FTP Data: Number of data bytes remaining = 65491 (0xFFD3)
00000:  00 02 FD 14 DF 50 00 40 05 35 D3 65 08 00 45 00   ..ý.ßP.@.5Óe..E.
00010:  05 DC 7D 35 40 00 80 33 E4 FF 0A 00 00 03 C0 A8   .Ü}5@. 3äÿ....À¨
00020:  C8 0E 06 04 00 00 1E DE 5A EE 00 00 5D 93 9A 6E   È......ÞZî..]“šn
00030:  57 D3 86 76 CD 18 08 AF 74 02 00 14 04 38 83 C4   WÓ†vÍ..¯t....8ƒÄ
```

```
00040:  41 39 ED 70 F2 97 50 10 44 70 BC F2 00 00 3F 57   A9ípò—P.Dp¼ò..?W

…

00590:  1B C1 49 E5 51 47 DD F4 FB 1F FA C4 BC 0F 7E 1C   .ÁIåQGÝôû.úÄ¼.~.
005A0:  8F 80 24 49 3C 18 F2 1A 70 63 79 7E 94 0A FE DD     $I<.ò.pcy~”.þÝ
005B0:  3F F0 C0 F9 41 94 44 0B C0 F9 29 4A 3C 00 40 A1   ?ðÀùA”D.Àù)J<.@¡
005C0:  FC C0 8A 15 04 9B BC BE CA 5F 75 CE EE C0 55 AB   üÀŠ..›¼¾Ê_uÎîÀU«
005D0:  1E 0C F9 0D 78 AA 7C 00 2B 56 15 88 3C 08 3C 56   ..ù.xª|.+V.ˆ<.<V
005E0:  BE 90 95 AB 0D 45 3E 05 9E 2A                     ¾ •«.E>. *
```

A.1.9 Test 5 – AH with NIC Encryption Engine

A.1.9.1 Hardware Setup

The two workstations, A and B, were configured with the onboard encryption NICs and set with the appropriate network configuration.

A.1.9.2 Software Setup

Windows 2000 was configured to establish an encrypted link between the two workstations using IPSEC AH using the AH/ESP Procedure. The Authentication Header option must be selected in steps 26 and 57. This was done for both workstations by enabling the NIC with the encryption engine, disabling the ordinary NIC, and then following the AH/ESP Procedure to create and enable the security policy requiring encryption.

Enable the NIC with the encryption engine, disable the conventional NIC, and enable encrypted communications by following the procedures outlined in section A.1.5.1.

The ASUS-based workstation running the Microsoft Network Monitor captured the frames of the file transfer.

A.1.9.3 Testing Procedure

The procedure specified in section A.1.5.2 was repeated.

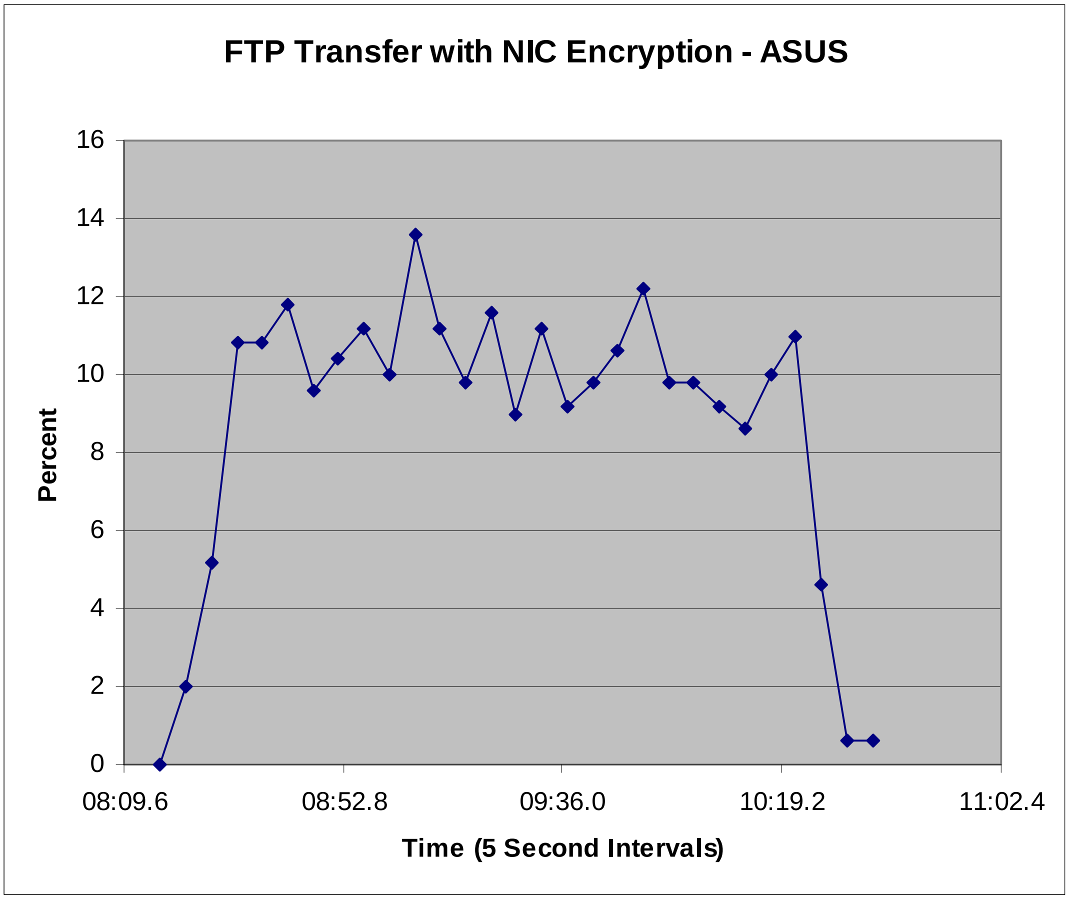
A.1.9.4 Expected Results

- The file transfer rate in kilobytes per second will be known.
- The CPU usage on the workstations will be known.
- The Microsoft Network Monitor output will verify that the IPSEC frames are properly formatted and the data is not in the clear.
- The IPSEC statistics will be known from the “ipsecmon” screenshots.
- Any anomalies will be documented.

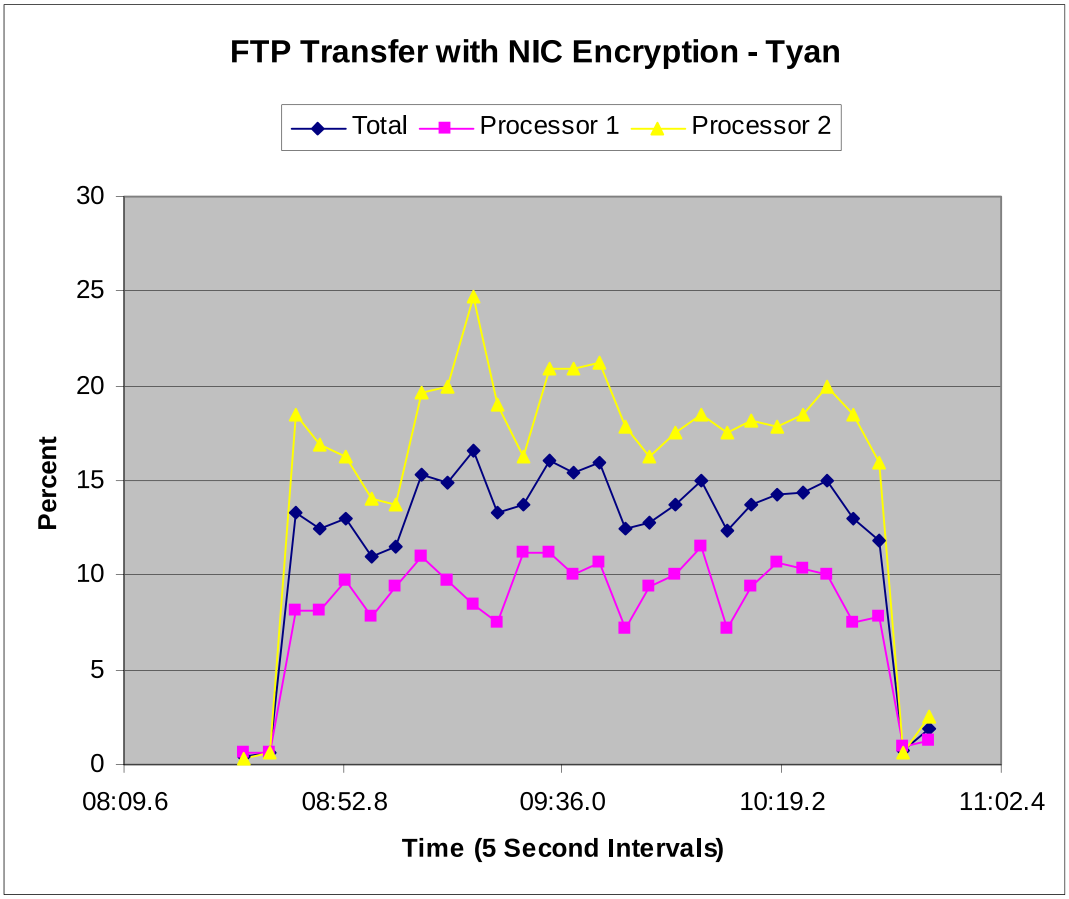
A.1.9.5 Results

The ftp transfer was for 106,278,016 bytes transferred in 118.52 seconds, for a rate of 896.74 kilobytes per second.

The ASUS CPU usage was as follows:



The TYAN CPU usage was as follows:



The total CPU usage of the TYAN-based system is calculated by adding the time both processors are used and dividing by the total time available. It effectively averages the CPU utilization since both processors have the same clock-rate.

Ipsecmon, the Windows 2000 Security Association monitor reported the following statistics:

IPSEC Statistic	TYAN	ASUS
Authenticated Bytes Sent	3,115,082	111,032,884
Authenticated Bytes Received	111,029,489	3,115,710
Key Additions	1	1
Oakley Quick Modes	1	1

The number of bytes varies probably due to additional Microsoft Windows activity. The Network Monitor, part of Windows 2000 Server, captured 1,10 AH packets for the session. The packets were identifiable as FTP packets.

A.2 SecurID: 2-factor Authentication

RSA's SecurID system provides the capability to authenticate the person based on both something they know (a pin or a password) and something they have (the SecurID token). It is a one-time password type of technology that generates a token once every 60 seconds. A user must have the token and a pin to authenticate.

The server runs as an internal server (behind a firewall) on the ground. The firewall provides one-time authorization to remote users, after contacting the SecureID/ACE server, so the only overhead is a one-time expense during user login. The server should be located on a protected network. Each user requiring a connection to a protected resource must have a SecurID device, usually a card or key fob, which generates the tokens, and in some cases a client. Each resource requiring protection must run an ACE agent. When the user attempts to connect to the protected resource, he or she is prompted for the token and pin. The agent communicates with the server to authenticate the user based on the token and pin.

A.2.1 CPU and Memory Utilization

The Cisco PIX firewall comes with a SecurID agent. The SecurID server was already configured on the Goddard Center Network Environment (CNE) network and could not be moved, because the license was linked to the host where it was installed, and the IP address of that host. Ordinarily, we would not want the server to be outside the firewall. Another host outside the firewall attempted to connect to the firewall using the SecurID token and pin. Approximately 20 to 25 tests were run to get a feeling for the variance of the results, and every attempt was made to reduce other background activity on the test machine. The data fluctuated a bit over numerous tests, but reasonable averages are given.

CPU utilization for the SecurID server was less than 0.1% running as a background process.

The memory usage was less than 25 megabytes. It is suspected that the program just grabbed up as much memory as it could find in anticipation of supporting large numbers of users. It is doubtful that the program actively uses 25 megabytes at any given time. Either way, in a system dedicated to authentication, this will not be an issue.

The memory and CPU utilization for the authenticating host and the firewall were not able to be measured. However, the SecurID authentication process was fast, and no devices in the test were overloaded or disrupted by the authentication. The only instance where the CPU and memory utilization for SecurID would be an issue is the instance of placing an agent or server on a spacecraft. This was not tested because we were unable to acquire an agent for Linux.

A.2.2 Latency or Delay

Latency is subject to the network condition, such that fast network connections would experience less delay than slow networks. The amount of traffic is very small, so collisions and re-transmissions should be minimal on a non-congested network. The delay caused by this authentication step is a number of round trip times between the client, the agent, and the server. It is a one-time event that takes a negligible amount of time in a ground network. To run SecurID on the spacecraft may be more complicated depending on the spacecraft distance and delay.

A.2.3 Bandwidth utilization and/or overhead

The SecurID authentication traffic is minimal, and so the bandwidth is also minimal. The messages consist of encrypted passwords, user IDs, 6-digit token, and 4-digit pins. The responses to the requests would be equally small. Once the authentication is complete, there is no bandwidth overhead or impact on subsequent communications.

A.2.4 Complexity or Ease of Use and Interoperability

Once configured, the system is very easy to use. From a client perspective, the client attempts to connect to the target machine. The client is prompted for a user ID and then a token/pin combination. The user may then be prompted for a login/application user ID and password. From a system administrator perspective, the server runs unassisted as a service on NT or 2000. Adding users is a matter of assigning a token card and pin, adding the user to the database, and entering the systems into the user's profile that he/she is allowed to access.

The server runs on NT and 2000. The clients can run on all flavors of Windows, and most flavors of UNIX. In addition, many firewalls and routers have built-in support for SecurID directly (known as ACE server support) or indirectly through a RADIUS server.

While the tool is proprietary, there is a large amount of support for it in the vendor community, and it does tend to integrate reasonably well with other technologies such as VPNs.

Initial setup tends to be more complex than necessary, and documentation a bit cryptic. However, once one understands the terminology and has a base configuration established, the addition of users and other sites is straightforward.

A.2.5 Features of Security

The SecurID authentication is 2-factor (something you have: token, and something you know: pin number). There is no encryption of the connection or application data; SecurID is merely an authentication tool. However, it does tend to integrate well into VPN scenarios. In fact, the usual scenario is to have a component in a firewall or router that establishes a VPN connection between the client and target host. The firewall also contacts the RADIUS or ACE server to authenticate the user. This provides a secure connection for login as well as application traffic.

The tool is based on the one-time password concept, considered the strongest login mechanism, and knowledge of a pin number. However, the passwords (i.e., the tokens) are based on some proprietary algorithm embedded in the token generation card. Since the server must also know the algorithm to predict the next token required of the user, the strength of the token generation is based on keeping the algorithm and the random seed numbers secure. Unlike true one-time passwords, which are completely random, tokens could be predicted if one knew the algorithm and token seed numbers. The RSA algorithm has been publicly disclosed. All token and pin information is encrypted if the transport is encrypted. Tokens are only good for 60 seconds maximum, maintained by synchronizing the server and token card clocks. Once a token is used, it cannot be immediately reused. The server can be configured to disable accounts after some number of failed login attempts. Users can be forced to change their pins on a scheduled basis. Further, the number of token numbers one would need to observe in order to use the algorithm to predict the next token number is extremely large. It is unlikely anyone has the time to mount this attack. SecurID is probably one of the most widely used electronic authentication systems available, but as with any mechanism or device, it is this system could conceivably be compromised.

A.3 Encryption Using SSH and IPSec

Client-to-Client IPSec VPN tunnels were tested to determine the feasibility of encrypting communications between a spacecraft and a ground host. IPSec was tested in two environments: the NSDNet lab, in GreenTec4, and the OMNI lab, in GSFC building 23. The NSDNet lab tests simulated solution concept 1. The OMNI lab tests simulated solution concepts 1, 2 and 3.

A.3.1 IPSec on OpenBSD Operating System: Solution Concept 1

Tests were done with SSH and IPSec (ESP) to test the cost of encryption on a 486 processor. These tests were conducted to provide some insight into the impact of encryption on a spacecraft CPU and bandwidth.

A.3.1.1 Capabilities Results Summary

Drax1, the 486 PC used as the simulated satellite, as configured is not the best platform for testing. Extremes in performance were noted throughout the tests. Temporarily replacing the 3Com 3C509 NIC with a 3Com 3C900 NIC showed a dramatic increase in performance on any test where Drax1 was not using 100% of the CPU. This showed that the 3C509 NIC was a limiting factor in the tests.

At this point, we have seen uplink speeds as low as 9,000 bytes per second using DES encrypted IPSec and as high as 102,000 bytes per second using Blowfish encrypted IPSec. Using scp, the best was 178,000 bytes per second. The Hubble Space Telescope, is an 80486DX2 with an external clock of 25MHz. It is not known whether Drax1's stated clock speed is internal or external. In either case, the Drax1 CPU is no faster than the HST CPU. HST has 1 Megabyte of RAM, and 1 Megabyte of EEPROM with seven wait states for read access. Data rates for HST are a downlink of 1 Megabit per second using single access, and an uplink of up to 32 Kilobytes per second. If these links used 8 bits per byte, a 486 processor performing data encryption is capable of keeping up with uplinks with no problems. A 486 processor may be capable of keeping up with downlink, but the CPU would have little or no time available for other processing.

A.3.1.2 Configuration

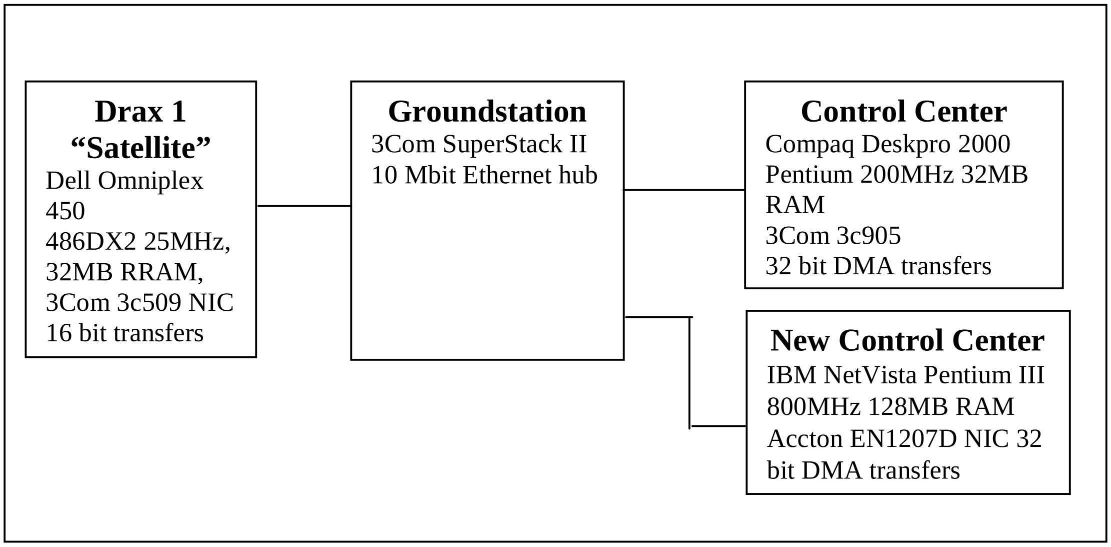


Figure A-2. Open BSD Simulated Satellite Test Configuration

Both Drax1 and the Control Center (CC) use OpenBSD 2.9 with IPSec and OpenSSH. The New Control Center (NCC) uses Windows 2000 SP 1. NCC was determined to have hardware problems during testing and was not replaced in time to perform encryption tests. This test configuration is equivalent to solution concept 1.

The Light Speed Tests and the Unencrypted Tests 1 to 5 were done using OpenBSD 2.8. A repeated sampling of tests showed no significant change in performance. An upgrade to OpenBSD 2.9 was performed because the IPSec implementation had some severe problems in version 2.8. Version 2.8 documentation was also less than ideal. There were no real improvements in IPSec documentation in OpenBSD 2.9.

The encrypted tests will determine the encryption performance of a 486 processor. Drax1 is intended to be a simulated spacecraft, however, it is not a perfect simulation. The simulated ground station does not add sufficient time delay. The hardware and OpenBSD were not designed for use in a real-time environment and this resulted in measurements that are not easily explained.

The baseline tests determined the speed of data transfers without using encryption. The tests of interest use encryption to protect the data. Both machines (Drax1 and CC) are in multi-user mode and each has normal background tasks that use a portion of the CPU. On Drax1 this portion is usually between 1% and 2%, and on CC it is less than 0.5%. These background tasks along with hardware timing issues impact testing because they add variability that would not exist on the spacecraft. For this reason, testing was limited to determining gross behavior. On tests where CPU utilization was over 95%, the results are rounded up to 100%, and other tests show the approximations or ranges of CPU utilization.

A.3.1.3 CPU Utilization

The simulated satellite CPU utilization for unencrypted file transfers was variable, ranging from 5 to 100% depending on what tool was used and which host initiated the transfer. However, in every case involving the simulated satellite sending encrypted traffic, CPU utilization was pegged at 100%.

A.3.1.4 Delay

Delay is added when encryption is used. Round trip time measured by a ping increased an average 6.2 milliseconds (ms) when IPSec with 3DES encryption algorithm was used. However, the delay was increased only by an average 5.3 ms when IPSec with DES or Blowfish encryption was used. The speed (and efficiency) of the encryption algorithm used does impact the bandwidth overhead and delay experienced by encrypted packets.

A.3.1.5 Maximum Bandwidth

The bandwidth capability (uplink and downlink) was affected by adding encryption. Inconsistent unencrypted baseline measurements again hinder an accurate depiction of the impact of encryption, however, an impact can be seen.

Table A-2. OpenBSD IPSec Throughput Degradation

		Transfer Rate (KBps)		
Direction	**FTP Initiator**	**Baseline**	**IPSec**	**Throughput Degradation**
Uplink	Ground	240	49	80%
Downlink	Ground	320	55	83%
Uplink	Spacecraft	50	49	3%
Downlink	Spacecraft	260	55	79%

A.3.1.6 Complexity or Ease of Use and Interoperability

Tests were entirely OpenBSD to OpenBSD for IPSec. Windows 2000 was abandoned due to problems with the test machine.

IPSec using manual keying is easy to do, but not all implementations support manual keying. There were some initial problems understanding the documentation. OpenBSD IPSec can use ISAKMP or Photuris for key management, but these methods were not tested.

SSH and SSL tunnels have the advantage of encrypting only selected traffic. This has the disadvantage of making traffic analysis easier to accomplish. There are a number of interoperability issues between the various implementations and versions of SSH. Like IPSec, the issue of key management is an unsolved problem for both SSH and SSL, but since this is largely a "people problem" it may never be solved by technological solutions.

A.3.1.7 Features of Security

IPSec (ESP) has the feature of encrypting all traffic between two points. This protects all traffic and reduces the effectiveness of traffic analysis. The disadvantage of encrypting all traffic is the performance hit if there is a large amount of traffic that is not sensitive. Subverting IPSec via external attacks is a very worthy challenge. Implementation mistakes and/or operating system bugs are possible weak points that will require vigilance in maintaining patches. The weakest part of any IPSec setup is the people who maintain it, but these sorts of problems will exist in any security system.

DES, 3DES, and AES are or will have NIST FIPS approval.

A.3.1.8 Light Speed Tests

Light Speed Tests (LST) are those that determine the maximum possible speed that an operation could achieve if all bottlenecks were removed. Reading from the device /dev/zero and writing to the device /dev/null should be the fastest possible read and write operations on UNIX systems. There are cases where this will not be true, but they do not exist in the hardware used for this test. The primary value in performing LSTs is to validate other tests. If for some reason, a test of some severe bottleneck shows a value approaching or exceeding the value from an LST, then that test is suspect. These tests were not run on NCC because dd, /dev/zero, and /dev/null do not exist on Windows 2000. Since NCC is so much faster than CC and Drax1 it is unlikely that any bottlenecks would be on NCC.

LST #1 /dev/zero to /dev/null

This test is to show the maximum possible speed that a system is capable of reading from one device and writing to another by only using the CPU. The command used was

```
  dd if=/dev/zero count=$BIGNUM > /dev/null
```

where $BIGNUM is large enough so that the test runs for at least 20 seconds. The speeds reported are those reported by the dd command. Drax1 has a speed for read and write of about 1.5 Million bytes per second, and CC has a speed for read and write of about 24 Million/bytes per second. This test has both machines at 100% CPU utilization.

LST #2 Disk Read and Write

This test is a repeat of LST #1 but replacing one of the devices with a file on the system hard disk. The commands used for read and write are:

```
  dd if=/tmp/zeros count=$BIGNUM > /dev/null

  dd if=/dev/zero count=$BIGNUM > /tmp/zeros
```

Drax1 is able to read about 600,000 and write about 430,000 bytes per second with 100% CPU utilization. CC can read about 7 million bytes per second (60% CPU) and write about 4 million bytes per second (50% CPU). Interrupt processing used about 33% of the CPU for reading and about 20% for writing on Drax1. On CC these figures were about 3% and 1%, respectively.

LST #3 Network Loopback Device

This test uses netcat to read from /dev/zero and write the results to the loopback network interface while a second netcat reads from that interface and writes to /dev/null. This determines the maximum possible speed that data can be passed through the network stack. Using the results of LST #1, the expected results cannot be greater than 1/3 those of LST #1. LST#1 used a single process to read and write the data once, and LST #3 uses three processes each reading and writing the data. The command used was:

```
  nc -l -p 1 > /dev/null &; dd if=/dev/zero count=$BIGNUM | nc -w 1 localhost 1
```

Drax1 and CC both showed about 1/3 of LST #1 and both used 100% CPU. This is an expected result. LST #1 reads the data once, and writes it once. LST #3 reads the data three times and writes it three times. This result shows that the network stack does not add significant overhead.

A.3.1.9 Unencrypted Tests

The following tests will be duplicated in one or more ways with encryption protecting the transmission. The results of the unencrypted tests are later compared with the results of the encrypted tests to determine the impact of encryption on bandwidth, delay, and CPU utilization on the simulated spacecraft.

UT #1 Netcat

This reads from a file on the transmitting machine and writes to another file on the receiving machine using netcat to pass the bytes over the wire. The machine that is listening uses the command:

```
  nc -l -p $PORT > /tmp/file
```

and the transmitter uses:

```
  dd if=/tmp/file count=$BIGNUM | nc -w 1 $HOST $PORT
```

With Drax1 listening the results were inconsistent. Speeds ranged from under 33,000 to 53,000 bytes per second with CC sending and 139,000 to 141,000 bytes per second with NCC sending. CPU utilization never achieved a steady state and Drax1's CPU utilization ranged between 5 to 25% with CC sending and from 46 to 76% with NCC sending. The CPU utilization was in the 1 to 2% range on CC and in the 1 to 5% range on NCC.

With Drax1 transmitting, speeds ranged from 250,000 to 350,000 bytes per second with 100% CPU utilization on Drax1. CC showed 6 to 9% CPU utilization and NCC showed 1 to 5%.

UT #2 Rcp or Rsh

Tests that write from CC to Drax1 have given results ranging from 100,000 to over 200,000 bytes per second and when Drax1 writes to CC, speeds range from about 160,000 to about 370,000 bytes per second. CPU usage on CC is usually in the 2 to 7% range, and on Drax1 it bounces wildly in the 20 to 100% range. Rcp and rsh do not exist on NCC.

UT #3 FTP

Transfers initiated on CC and puts from Drax1 use 100% of Drax1's CPU and 4 to 10% of CC's CPU. When Drax1 initiates a get, it uses only 10 to 28% of the CPU, and CC uses 1 to 2% and the transfer rate is about 50,000 bytes per second. Drax1's put sends about 260,000 bytes per second. CC's put and get operations transfer about 240,000 and 320,000 bytes per second. NCC does not have an FTP daemon. Puts and gets from NCC used 100% of Drax1's CPU and had rates ranging from 213,000 to 225,000 bytes per second with 1 to 7% CPU utilization on NCC.

UT #4 TFTP

All transfers were in the 130,000 to 139,000 bytes per second and used 100% of Drax's CPU. CPU utilization was 0 to 7% on NCC and 5 to 13% on CC.

UT #5 Ping Timing

This test uses the ping command to get an estimate on the minimum latency on sending a packet and getting a response. Drax1 pings itself with an average round trip time just under 3 milliseconds (ms). CC pings itself with an average round trip under 0.3 ms. Most packets are less than the average as roughly 10 percent of packets have a significantly longer round trip than the rest. For example, in the case of CC pinging itself, most packets make the round trip in less than .23 ms and the rest are more than .6 ms. The roundtrip when CC pings Drax1 averages 1.7 ms, which is much quicker than Drax1can ping itself. Initiating ping on Drax1 to NCC or CC, the roundtrip average is about the same as having Drax1 ping itself. Ping on NCC is only accurate to the nearest 10ms and did not yield usable results. Pinging NCC from Drax1 showed an average of 2.7ms, and from CC an average of .6ms.

A.3.1.10 Encryption Tests

ET #1 Rcp Disk to Disk

Using 3DES (the default encryption) the transfer rate is about 18,000 bytes per second in all cases. Drax1 CPU utilization is 100% and CC ranges from 5 to 8%. Using Blowfish encryption, we start to see some of the instability seen in rcp. Transfers that send data to Drax1 ranged from 39,000 to 43,000 bytes per second and CPU usage on Drax ranged from 17 to 55% while CC ranged from 2 to 7%. Transfers to CC were about 150,000 bytes per second, with Drax1 using 100% of the CPU, and CC has CPU utilization of 13 to 16%. Automating the test and repeating it with the SSH Protocol Version 2 encryption algorithms after the upgrade to OpenBSD 2.9, the following results were obtained (without CPU utilization):

Table A-3. Bandwidth Impact of Encryption Algorithms

Encryption Algorithm	Downlink Bytes/second Transmitted	Uplink Bytes/second Transmitted
aes128-cbc	83,000 to 128,000	14,000 to 18,000
aes192-cbc	84,000 to 112,000	14,000 to 18,000
aes256-cbc	83,000 to 97,000	14,000 to 17,000
blowfish	99,000 to 152,000	14,000 to 18,000
cast128-cbc	102,000 to 146,000	14,000 to 17,000
arcfour	161,000 to 178,000	15,000 to 17,000
3des	18,000 to 19,000	18,000
3des-cbc	15,000 to 16,000	10,000 to 12,000

ET #2 IPSec DES (Drax1 & CC)

Pinging with 1 packet per second, CC saw an average round trip to Drax1 of 7.4 ms, an increase of 5.7 ms, and Drax1 saw round trips of about 8 ms, a bit more than 5 ms longer than without IPSec. Uplink speeds for netcat, rcp, and FTP was in the 9,000 to 12,000 bytes per second range, and downlink was in the 72,000 to 88,000 bytes per second. Drax1 used 100% of the CPU for downlinks and 9 to 24% on uplinks. CC used 7 to 18% on downlinks and 1 to 4% uplink. TFTP speeds were 38,000 to 44,000 bytes per second. CPU utilization on Drax1 was 88 to 96% and CC was 7 to 15%.

ET #2 IPSec 3DES (Drax1 & CC)

Pinging with 1 packet per second, CC saw an average round trip to Drax1 of 8.6 ms, an increase of 6.9 ms, and Drax1 saw round trips of about 8.8 ms, an increase of 5.5ms than without IPSec. Uplink speeds for netcat, rcp, and FTP was in the 40,000 to 57,000 bytes per second range, and downlink was in the 54,000 to 56,000 bytes per second. Drax1 used 100% of the CPU on all transfers and CC used 12 to 25% of the CPU. TFTP transfers were about 32,000 bytes per second. CPU utilization on Drax1 was 82 to 94% and CC was 5 to 19%.

ET #2 IPSec Blowfish (Drax1 & CC)

Pinging with 1 packet per second, CC saw an average round trip to Drax1 of 7.5 ms, an increase of 5.8 ms, and Drax1 saw round trips of about 7.8 ms, about 4.8 ms longer than without IPSec. Uplink speeds for netcat, rcp, and FTP was in the 10,000 to 13,000 bytes per second range, and downlink was in the 95,000 to 102,000 bytes per second. Drax1 used 100% of the CPU for downlinks and 8 to 18% on uplinks. CC used 10 to 15% on downlinks and 1 to 3% uplink. TFTP speeds were 44,000 to 48,000 bytes per second. CPU utilization on Drax1 was 88 to 94% and CC was 8 to 16%.

A.3.2 IPSec on BlueCat Operating System: Solution Concepts 1, 2, & 3.

This section will discuss in detail the results of testing IPSec on a BlueCat Embedded operating system. BlueCat is a non real-time embedded Linux operating system created by Linux Works.

A.3.2.1 Test Configuration Description

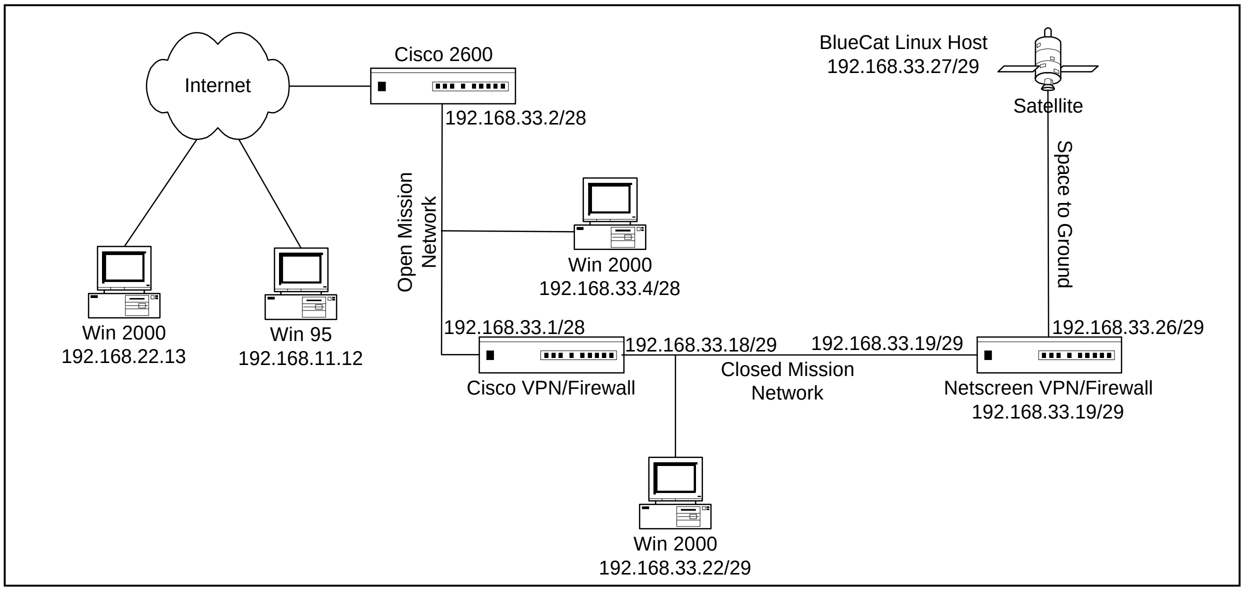


Figure A-3. Prototype Configuration: BlueCat Concepts 1, 2, and 3

Figure A-3 shows the test configuration. The BlueCat system is as close to what is flown today as possible: 486Mhz (Clock Speed of 66Mhz), 16MB RAM, ~110MB Disk space, 256 KB cache, and math coprocessor is installed. The Cisco firewall is a PIX 515 with 64 MB of memory. The Windows2000 machines are Pentium II/266 with 192MB. The Netscreen is a 5XP. All the connections except the internet are on 10Mb Ethernet.

FreeS/WAN version 1.7, used in this prototype, is an open source implementation of IPSec on Linux and BSD platforms. Pluto is the daemon used by FreeS/WAN for IPSec phase 1, the ISAKMP key negotiation. The Cisco VPN/Firewall protects the Closed Mission Network (192.168.33.16/29) from the Open Mission Network. The PIX is configured to allow the setup of IPSec tunnel from any of the machines shown on the Open Mission Network or Internet side. One of the Windows 2000 machines uses the native IPSec capability and configured as a gateway-to-gateway configuration to setup an IPSec tunnel between itself and the PIX firewall. The other Windows 2000 machine uses Cisco VPN client 3.0 to establish an IPSec tunnel between itself and the PIX firewall. The Windows 95 machine uses the Cisco Secure VPN Client 1.1 to establish an IPSec tunnel between itself and the PIX firewall. The setup on each of the machines requires an IPSec tunnel whenever they want to access any node on the 192.168.33.16/29 network. Any machine that establishes an IPSec tunnel to the PIX is allowed access to any node on the 192.168.33.16/29 network after the user is authenticated by the SecurID server. The SecurID server normally should be placed on the trusted network behind the firewall. However, the test configuration used a SecurID server that was on the 192.168.22.0 network. This server could not be moved from its networks segment because its license was linked to its IP address. The SecurID server agent that is enabled on the PIX firewall authenticates users by interfacing with the SecurID server. Any machine on the Closed Mission Network is allowed access to any machine on the Open Mission Network or Internet. The configuration file used on the PIX is given in Appendix E. The details of various client configurations on Windows machines are given in Appendices F-H. The Netscreen VPN/Firewall (Netscreen 5) is configured to allow the set up of an IPSec tunnel from the BlueCat Linux host, which is the simulated spacecraft. Communication is allowed between the BlueCat and any machine on the 192.168.33.16/29 and the 192.168.33.0/28 subnets, and the machines with the IP addresses of 192.168.22.13 (Windows 2000), 192.168.11.12 (Windows 95 machine), and 192.168.22.15 (DNS). This required the setup of five separate tunnels between BlueCat and the Netscreen VPN/Firewall (two for each subnet and three for each specific machine). The DNS was required for the ftp and some other applications that were installed on the BlueCat. In a true spacecraft system implementation, DNS access should not be required by the spacecraft. The Netscreen firewall also allowed any machines on the Closed Mission Network to access machines on the Space-to-Ground network. The BlueCat configuration file is shown in Appendix C. The Netscreen configuration is shown in Appendix D.

All the IPSec tunnels used ESP with 3DES encryption and SHA for authentication. ISAKMP used Diffie-Hellman Group 2, SHA, and pre-shared secrets. The ISAKMP Security Associations (SAs), or tunnels, may use Oakley groups MODP1024 (Diffie-Hellman group 2) and MODP1536 (Diffie-Hellman group 5); 3DES encryption; SHA1 and MD5 authentication. The IPSec SAs (tunnels used in IPSec phase 2 – the actual data communication) may use 3DES and MD5 or SHA1 for ESP, or just MD5 or SHA1 for AH. IPCOMP Compression, a header compression feature, is always 'Deflate,' or disabled.

For all test purposes the only daemons that were utilized were inetd, tcpd (FTP/Telnet applications), sshd (for SCP, SFTP, and SSH version 2.0), whackd (control interface for Pluto) and Pluto (for IKE's ISKAMP key negotiation protocol).

A.3.2.2 CPU and/or Memory Utilization

The throughput and CPU utilization on various nodes were measured by using FTP to transfer a 15MB file between various nodes. A Baseline was taken by directly connecting the BlueCat and a Windows 2000 host, with no firewall or VPN tunnel between them. Concept 1 was simulated by the FTP between the BlueCat and the Windows 2000 host 192.168.33.22. In this case, a VPN tunnel was set up between the netscreen firewall and the BlueCat system. Concepts 2 and 3 were simulated by the FTP between the BlueCat and the Windows 2000 host 192.168.22.4. In the concept 2 & 3 test, VPN tunnels were created between the BlueCat and the Netscreen firewall, and between aWindows 2000 host and the PIX firewall. In each test, a binary ftp get and put were executed from both the BlueCat and the Windows hosts. The tests were repeated and averages are reported.

Memory Utilization

The memory utilization for the daemons are as follows:

Pluto/Wackd = 3.7% and 5.3% for Initialization
Inetd = 0.7%
In.ftpd = 3.3% for sending and 5.9% for receiving

These are all of the daemons utilized in the security configuration. IPChains runs in the same user space as the Linux kernel and uses the same resources as the kernel would i.e. the Linux operating system.

CPU Utilization

Table A-4. BlueCat and Windows IPSec CPU Utilization Results

	BlueCat to Windows BlueCat Initiated	BlueCat to Windows Windows Initiated	Windows to BlueCat BlueCat Initiated	Windows to BlueCat Windows Initiated
Windows CPU Utilization				
Baseline	100%	20%	100%	5%
Concept 1	100%	7%	100%	3%
Concepts 2 & 3	100%	20%	100%	14%
BlueCat CPU Utilization				
Baseline	45%	40%	48%	80%
Concept 1	96%	98%	30%	45%
Concepts 2 & 3	96%	97%	30%	30%
FTP Transfer Rate (KB/s)				
Baseline	610	580	440	650
Concept 1	120	140	64	120
Concepts 2&3	130	140	67	114

The variation in CPU utilization reported was within 5% throughout test trials. During the initialization stages of the IPSec tunnel, 80% of the CPU is utilized. Once the tunnel is up, the CPU utilization is affected merely by the encryption of IPSec.

In Figure A-4, the BlueCat CPU Utilization is plotted. It is clear that the CPU utilization is high when the BlueCat host is reading data from disk, encrypting it, and transmitting it. However, when it is receiving encrypted data, decrypting it, and writing it to disk, the CPU is not used as heavily. This would imply that it is harder to encrypt than to decrypt. However the 3DES algorithm is run the same way to encrypt and decrypt, so the CPU should be performing the same process in either case. The baseline data shows that the BlueCat does not spend more CPU resources to read data and send it than to receive data and write it. Extensive testing may be required to determine the cause of this asymmetric CPU utilization. However, the same asymmetry exists in the baseline FTP tests

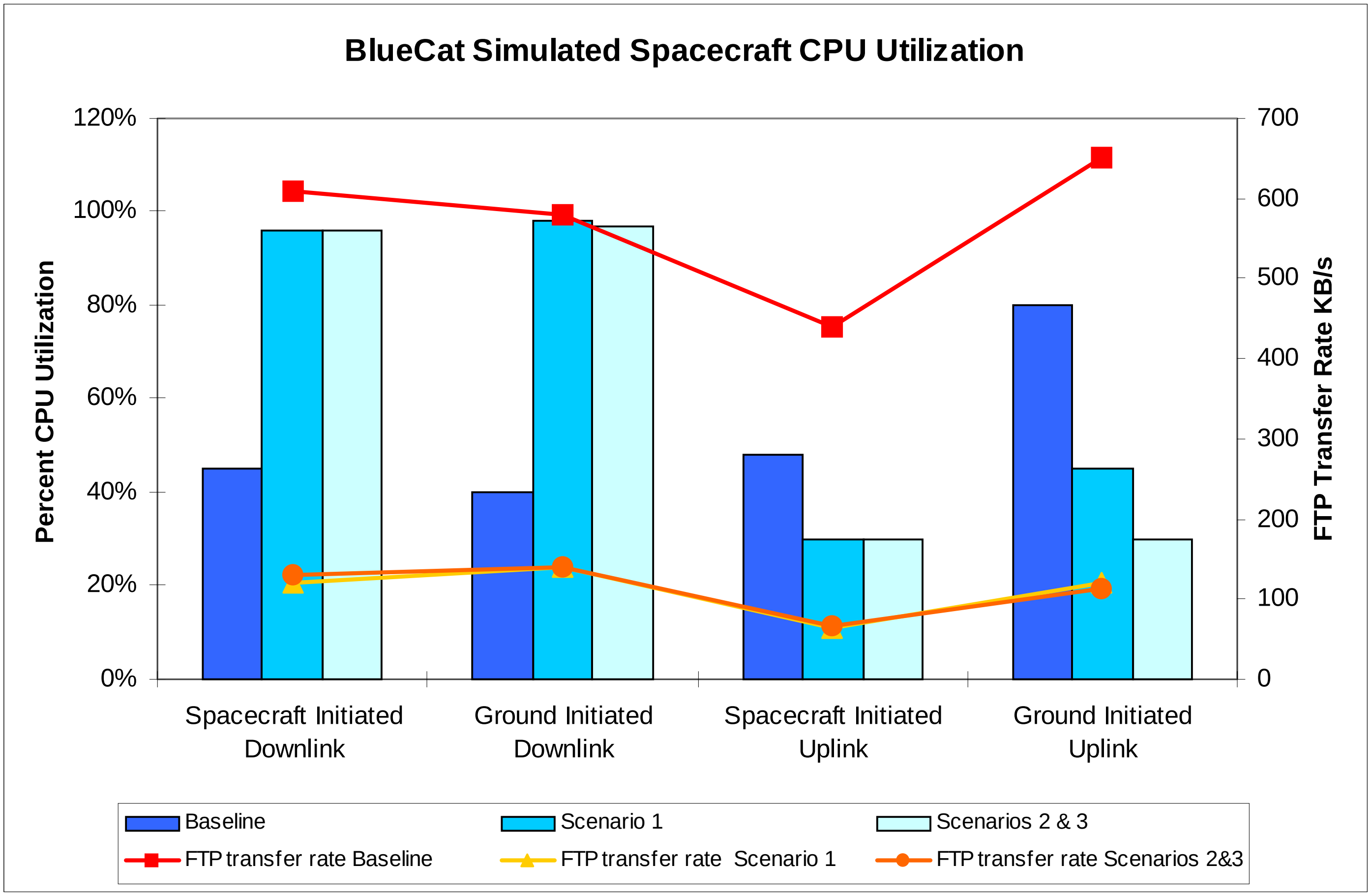


Figure A-4. BlueCat IPSec CPU Utilization

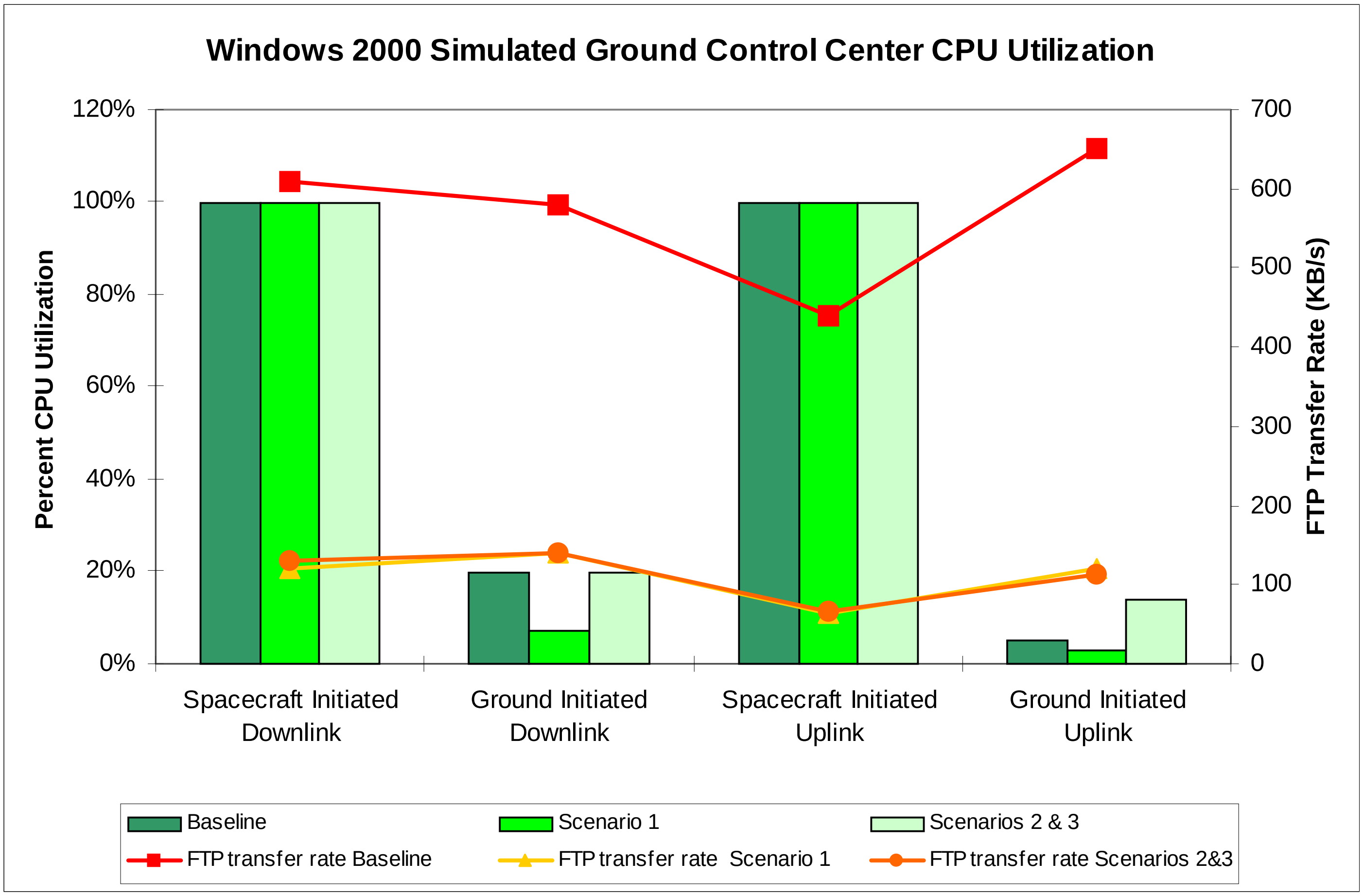


Figure A-5. Windows 2000 IPSec CPU Utilization

A.3.2.3 Latency or Delay

Pluto takes 15 seconds for initialization and 3 seconds thereafter for each IPSec tunnel instantiated. Once the tunnel is established, the only added delay is due to less effective throughput.

A.3.2.4 Bandwidth utilization and/or overhead

The throughput degradation due to adding IPSec was an average 80.6%.

Table A-5. BlueCat IPSec Throughput Degradation

		Transfer Rate (KBps)			**Throughput Degradation**	
Direction	**FTP Initiator**	**Baseline**	**Concept 1**	**Concepts 2& 3**	**Concept 1**	**Concepts 2& 3**
Downlink	Spacecraft	610	120	130	80%	79%
Downlink	Ground	580	140	140	76%	76%
Uplink	Spacecraft	440	64	67	85%	85%
Uplink	Ground	650	120	114	82%	82%

A.3.2.5 Complexity or Ease of Use and Interoperability

The tests used IPSec implementations from Cisco, Netscreen, Microsoft, and FreeS/WAN. Since each one of them offers various key exchange techniques, encryption and authentication algorithms, a common set of parameters that works on all of the products needs to be selected. When designing a secure system, all the potential IPSec products should be studied closely to determine that they all support the desired encryption algorithms and key exchange mechanisms.

FreeS/WAN will work on any implementation of Linux. FreeS/WAN is very easy to configure with pre-shared keys or in a key server distribution environment. The Diffie-Hellman Groups 2 and 5 are supported by Pluto. FreeS/WAN will not support Group 1 or 3 negotiation.

FreeS/WAN is the only freely available implementation for Linux that is mature and works with Cisco, and Netscreen’s VPN. Although not yet tested, it probably would be compliant with Windows 2000 IPSec client as well, considering that Windows 2000 was compliant with Cisco and Netscreen VPN devices.

The FreeS/WAN VPN tunnel setup is simple once one understands the concept and terminology used in the configuration files. The ***ipsec.conf*** and ***ipsec.secrets*** files in the /etc directory are used to configure FreeS/WAN. The ***ipsec.secrets*** file defines the shared secrets used between VPN endpoints during the ISAKMP portion of IPSec. The ***ipsec.conf*** file defines the tunnels, or Security Associations (SA), that are considered acceptable and that may be set up by IPSec phase 1 to communicate data in IPSec phase 2. The ipsec.conf file used in these tests is displayed in Appendix C.

Pluto, the daemon that negotiates and sets up the IPSec tunnel, will negotiate with any non-FreeS/WAN IPSec implementation, from lowest level of security to highest. In other words the Pluto would try to establish SAs using MAC (Message Authentication Code) MD-5 and DES before it tried SHA-1 and 3DES. FreeS/WAN then uses the first SA match that it finds during the negotiation phases with the other IPSec device, *regardless* of what is configured as acceptable in the ipsec.conf file. Therefore, the least secure SA that is acceptable to the non-FreeS/WAN IPSec device (and one which may not be acceptable according to the ipsec.conf file) is accepted by FreeS/WAN and used for IPSec phase 2 communication. The latest version of FreeS/WAN, version 1.9, does not correct this flaw. The allowable IP addresses specified in the ipsec.conf file are used. FreeS/WAN will not set up a tunnel with another VPN device if that device’s IP address is not specified in the ipsec.conf file. The ipsec.conf file, including SA parameters is followed correctly when a tunnel is established from one FreeS/WAN host to another FreeS/WAN host. This is a severe interoperability issue.

The Cisco Secure VPN Client 1.1 configuration is easily managed through a GUI. This client does not work on Windows 2000. Cisco VPN Client 3.0 works on all Windows machines. However, this client requires the Firewall/VPN-Concentrator to assign it a dynamic IP address from a pool of IP addresses. Cisco provides a client for Linux but does not provide a client for UNIX. Solaris 8 has IPSec capabilities. However, it does not support ISAKMP.

Windows 2000 allowed for the 3DES option in its configuration screen. However, when the tunnel was initiated, it did not use 3DES. This problem with Windows 2000 is fixed by downloading and installing the *'Strong Encryption'* option for Windows 2000.

Each of the products was relatively easy to use in and of itself. They each implement IPSec in a different way, and specify different ways of configuring the IPSec tunnels. Integrating them together was possible, but took some extra time to learn each of the different products, to determine how they were interacting with each other, and to set up each piece the correct way.

A.3.2.6 Features of Security

BlueCat has the same software architecture as Red Hat Distribution 6.0, but has modifications in certain daemons and services that help improve security capabilities. BlueCat allows the kernel and file system to be built from scratch, which was done in this prototype. This provides the benefit of having only the needed capabilities and nothing more within the operating system.

FreeS/WAN follows the RFCs defined by IETF and executes the algorithms set forth by the VPN Consortium. FreeS/WAN has the ability for you to configure RSA key in secure DNS servers, utilization of PKI servers instead, or the use pre-shared key methods. Pre-shared keys, which are located in the ipsec.secrets file, were used in these experiments. Ideally, for less than five PIs or FOTs that would have access to the spacecraft, this would be sufficient. With more users, key exchange servers may be needed to facilitate key management.

Methods of key management were not addressed in this effort. Factors that complicate key management are the physical inaccessibility of spacecraft after launch, and the broadcast nature of the RF link which might be used to communicate keys or other information used to create or choose keys.

In a study submitted for the IACR (International Association for Cryptologic Research), the IPSEC standard was portrayed to be more secure than SSH and SSL.

Digital Signature Algorithm (DSA), DES, 3DES, and SHA are certified by NIST as FIPS compliant.

The IPSec specification is under the IETF authority. The IPSec specification consists of numerous documents. The most important of these, issued in November of 1998, are RFCs 2401, 2402, 2406 and 2408. The data encryption standard (DES) and secure hash algorithms (SHA) are developed by NIST. FIPS PUB 46-3, FIPS PUB 180-1, and FIPS PUB 186-1 provides details on DES, 3DES, SHA-1, and DSA.

A.3.3 Firewalls: IPChains, Cisco, and Netscreen

This section documents the firewall capabilities of IPChains, the Cisco PIX-515 and the Netscreen 10 commercial firewalls. The test configuration with these two firewall/VPN devices is shown in Figure A-3.

IPChains was included in the prototype as a stateless firewall onboard the spacecraft. IPTables, which is a stateful firewall and improved over IPChains, only runs on kernel versions 2.4.X, which is an unstable version of Linux and currently not supported by most embedded systems architectures. The BlueCat kernel version used in the prototype is 2.2.12-1. Therefore, the prototype was unable to run IPTables, and IPChains version 1.3.9 was used instead.

A.3.3.1 Test Configuration

The configuration of the firewall tests has been detailed in section A.3.2.1. In the tests described in section A.3.2, observations of firewall performance were measured.

A.3.3.2 CPU Utilization

IPChains is patched in the kernel source and compiled statically. It is not capable of being loaded dynamically as a module at runtime. Therefore, it uses the same user space and resources as the kernel in order to operate.

The tests described in section A.3.2 on IPSec were run without IPChains, and then repeated with IPChains. The sets of tests showed no measurable difference in CPU utilization, memory utilization, or latency. Therefore, IPChains may not add substantial resource requirements. The IPChains rule set was not extensive, and a lengthy rule set may contribute to a noticeable performance difference. We recommend a short rule set that allows only a small number of IP addresses and ports and denies all other traffic. Such a rule set will be more secure and will use fewer resources.

The CPU utilization for the Cisco PIX and Netscreen firewall VPN gateways was measured in the tests described in section A.3.2. In addition, the equivalent of Concept 1 was run for the PIX, using two Windows 2000 hosts on either side of it to FTP data. FTP was initiated from a PC on the Open Mission Network (Windows A) to a PC on the Closed Mission Network (Windows B). The results are shown in Table 4-4.

Table A-6. Firewall/VPN Gateway CPU Utilization

Concepts 2 & 3	**BlueCat to Windows BlueCat Initiated**	**BlueCat to Windows Windows Initiated**	**Windows to BlueCat BlueCat Initiated**	**Windows to BlueCat Windows Initiated**
Netscreen CPU Utilization	8%	10%	6%	10%
PIX CPU Utilization	20%	20%	13%	19%
FTP transfer rate (KB/s)	130	140	67	114
Concept 1	**BlueCat to Windows BlueCat Initiated**	**BlueCat to Windows Windows Initiated**	**Windows to BlueCat BlueCat Initiated**	**Windows to BlueCat Windows Initiated**
Netscreen CPU Utilization	11%	10%	6%	12%
FTP transfer rate (KB/s)	120	140	64	120
Concept 1	**Windows A to Windows B Windows A Initiated**	**Windows B to Windows A Windows A Initiated**		
PIX CPU Utilization	72%	72%		
FTP transfer rate (KB/s)	575	450		

The firewall CPU utilization varied with the throughput of the FTP transfer. On average, the Netscreen firewall/VPN gateway was not taxed by the amount of data flowing to and from the BlueCat host (max 140KB/s). Its CPU utilization never went above 12%. The PIX firewall was also not taxed by the amount of traffic flowing to and from the BlueCat host, with about 20% utilization. However, when two Windows 2000 hosts communicated through the PIX at a much higher rate (~500 KB/s), the PIX had to use more than half its CPU to handle the load. Clearly, in the ground scenario, firewall technology is mature enough that firewalls could be expected to easily keep up with data rates of satellite communication.

A.3.3.3 Bandwidth Utilization and Overhead

Firewalls do not add any bandwidth. They may behave as bottlenecks and decrease the rate at which traffic can pass. However, such an effect was not measured in these tests.

A.3.3.4 Ease of Use and Interoperability

The PIX firewall was configured using the Command Line Interface (CLI). The Netscreen was mainly configured using the web interface. It is easier to use web interface than the CLI. The PIX Device Manger (PDM) that is provided in release 6.0 of PIX software allows the configuration of PIX using a web interface. Debug and log message available on the Firewalls and the clients aids the configuration.

IPChains runs only in Linux kernels 2.2.14-X. For Red Hat 7.X or kernel 2.4.X and later, IPTables is used instead. IPChains rules will only work with Linux versions of IPChains. IPChains has only been implemented with Linux as well as IPTABLES.

IPChains configuration tags are cryptic and mistakes in the configuration can result in a false sense of security for the system. The *PMFIREWALL* script was utilized for configuring IPChains initially to avoid some of these problems. Then the configuration files were modified by hand to suit the prototype needs.

IPChains defines three chains: input, forward, and output. All packets whose destination address is the host running IPChains enter the Input Chain. The Forward Chain is for packets traversing through the host, but not originating or terminating at the host. The Output Chain is for packets originating from the host running IPChains. Each chain has rules that allow or deny packets based on port or protocol information.

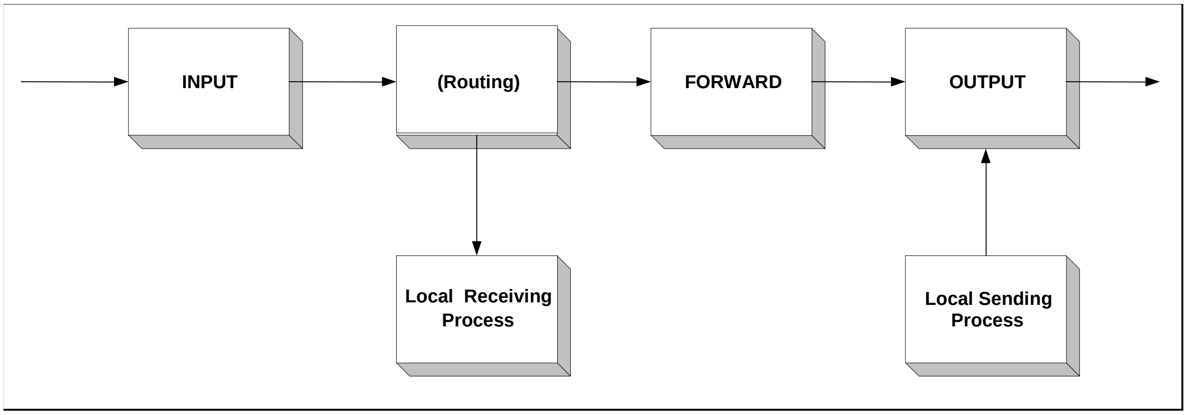


Figure A-6. IPChains Packet Filtering Architecture

The IPChains architecture diagram (figure A-6) shows that input and output packets only go though the input and output chains, respectively, but that all through packets go through the input, forward, and output chains. This architecture is very important to understand so that the policy or rule set is configured correctly. For example, if traffic is blocked in the input chain rules but permitted in the forward chain rule, the traffic would be blocked, because the traffic would never reach the forward chain and the rule permitting it to pass.

A.3.3.5 Features of Security

IPChains provides static packet filtering for ingress and egress traffic of the host or for through traffic (if the host is configured as a router). It can filter based on IP addresses, IP subnet, other IP, TCP, UDP, or ICMP header fields, and port. IPChains can log connection data, and has support for NAT. However, IPChains will not handle most denial of service attacks as IPTables could. IPTables, like most stateful firewalls, has intelligence to automatically block addresses that are mounting a denial of service attack. IPChains lacks that capability. Instead, the administrator has to view logs to determine which addresses to block manually.

The same consortium responsible for creating Linux develops IPChains. The SANS Institute recommends IPChains and IPTables.

Appendix B. IPSec Overview

IPSec was defined by RFC 2401 in November 1998. It is a standard method to protect network traffic from unauthorized access as it passes over open networks. IPSec offers the following:

- Data Origin Authentication
- Data Integrity
- Replay Protection
- Data Confidentiality.

IPSec uses a two-phase process. Phase 1 is used to establish keys and conditions to be used in the communication of Phase 2. Phase 2 is the transmission of an actual protected communication.

The specified public-key based approach for Phase 1 uses IKE: Internet Key Exchange, RFC 2409. IKE is based on ISAKMP key exchange framework and Oakley and SKEME key exchange techniques. ISAKMP, KDC-based systems such as Kerberos, or other systems such as SKIP may also be used in Phase 1.

Phase 1 first authenticates both communication endpoints, and establishes a secure communication between them. Signatures, public-key encryption, and pre-shared key may be used for authentication.

The *security association* (SA) is also set up in Phase 1. A security association contains protocol and encryption information, including encryption keys, to be used in future exchanges of Phase 2. The communication initiator proposes a SA, and the responder either accepts it, or proposes a different SA. Eventually both ends must agree on the SA for IPSec Phase 2 to commence. The key exchange portion of establishing the SA is usually done with the Diffie-Hellman algorithm. This algorithm is a method for a shared secret key to be created using information passed in public. Although this step may be done in the clear, many implementations allow ISAKMP exchanges to be encrypted at this stage using pre-shared keys, or public-key encryption.

IPSec Phase 2 is the actual data transmission phase. There are two protocols for IPSec Phase 2: Authentication Header (AH), and Encapsulating Security Payload (ESP). The AH protocol provides authentication, integrity, and replay protection for the entire packet, including the IP headers. The ESP protocol provides authentication, integrity, and confidentiality for the IP datagram. In addition, there are two modes for IPSec: tunnel and transfer. The tunnel mode encapsulates and protects (according to the specified protocol) the entire packet, including source and destination IP addresses. This mode adds a new IP header, with the source and destination IP addresses listed as the endpoints of the VPN tunnel. Transfer mode uses the original packet header information, so the true source and destination host IP addresses are visible in the packet headers. Tunnel mode is most often used.

The following table illustrates which portions of an IP packet are protected in the combinations of the two modes and two protocols.

Table B-1: IPSec Packet Construct

PACKET CONSTRUCT	Authentication Header	Encapsulating Security Payload
Transfer Mode	*IP Header 1* *AH Header* *DATA*	IP Header 1 *ESP Header* ***DATA*** ***ESP Trailer*** ESP Authenticator
Tunnel Mode	*IP Header 2* *AH* *IP Header1* *DATA*	IP Header 2 *ESP Header* ***IP Header 1*** ***DATA*** ***ESP Trailer*** ESP Authenticator

- Italic denotes authenticated portions of a packet.
- Bold denotes encrypted portions of a packet.
- IP Header 1 is the original IP header information in a packet.
- IP Header 2 is the IP header added by IPSec, including the source and destination addresses listed as the endpoints of the IPSec tunnel.
- AH Header is header information added by the AH protocol.
- ESP Header, ESP Trailer, and ESP Authenticator are headers and trailers added by the ESP protocol.

Appendix C. IPSec and SSH Configuration Files on BlueCat

The ***ipsec.conf*** and ***ipsec.secrets*** files in the /etc directory are used to configure IPSec on BlueCat. The ***ipsec.secrets*** file defines the shared secrets used between BlueCat and Netscreen during the ISAKPMP. The ***ipsec.conf*** file defines the tunnels that are setup at the start of IPSec. The contents of these files are shown on the next page.

The ipsec.conf file shown on the next page defines setup for five security associations (SA) for five different connections. The first one named Trusted_Net defines the connection from BlueCat to any node in the trusted segment (192.168.33.16/29). The one named Untrust defines the SA from BlueCat to any node on the untrusted (192.168.33.0/28) segment. The connections named W_2K, green_tec, and dns define the SAs to specific machines from BlueCat. Note that the tunnel exits only from the BlueCat to the Netscreen with five different SAs. The definition for the DNS was required because the BlueCat needed to contact the domain name server (DNS) when executing network applications like the FTP.

Ipsec.conf

```
# /etc/ipsec.conf - FreeS/WAN IPSEC configuration file

# More elaborate and more varied sample configurations can be found
# in FreeS/WAN's doc/examples file (http://freeswan.org/doc.html).

# basic configuration
config setup
      # THIS SETTING MUST BE CORRECT or almost nothing will work;
      # %defaultroute is okay for most simple cases.
      #interfaces=%defaultroute
      interfaces="ipsec0=eth0"
      # Debug-logging controls:  "none" for (almost) none, "all" for lots.
      klipsdebug=none
      plutodebug=none
      # Use auto= parameters in conn descriptions to control startup actions.
      plutoload=%search
      plutostart=%search
      # Close down old connection when new one using same ID shows up.
        #uniqueids=yes

# defaults for subsequent connection descriptions
conn %default
      # How persistent to be in (re)keying negotiations (0 means very).
#     keyingtries=1
      # Parameters for manual-keying testing (DON'T USE OPERATIONALLY).
      # Note:  only one test connection at a time can use these parameters!
#     spi=0x200
#     esp=3des-md5-96
#     esp=3des-sha1-96
#     espenckey=0x01234567_89abcdef_02468ace_13579bdf_12345678_9abcdef0
#     espauthkey=0x12345678_9abcdef0_2468ace0_13579bdf
      # RSA authentication with keys from DNS.
       #authby=rsasig
#     leftrsasigkey=%dns
#     rightrsasigkey=%dns
# sample connection
conn Trusted_Net
      keyingtries=1
      left=192.168.33.27
      right=192.168.33.26
      rightsubnet=192.168.33.16/29
      # To authorize this connection, but not actually start it, at startup,
      auto=start

conn W_2K
      keyingtries=1
      left=192.168.33.27
      right=192.168.33.26
      rightsubnet=192.168.22.13/32
      # To authorize this connection, but not actually start it, at startup,
      auto=start

conn green_tec
      keyingtries=1
      left=192.168.33.27
      right=192.168.33.26
      rightsubnet=192.168.11.12/32
      # To authorize this connection, but not actually start it, at startup,
      auto=start

conn dns
      keyingtries=1
      left=192.168.33.27
      right=192.168.33.26
      rightsubnet=192.168.22.0/0
      # To authorize this connection, but not actually start it, at startup,
      auto=start

conn Untrust
      keyingtries=1
      left=192.168.33.27
      right=192.168.33.26
      rightsubnet=192.168.33.0/28
      # To authorize this connection, but not actually start it, at startup,
      auto=start
```

ipsec.secrets

```
# This file holds shared secrets or RSA private keys for inter-Pluto
# authentication.  See ipsec_pluto(8) manpage, and HTML documentation.

# Shared secret (an arbitrary character string, which should be both long
# and hard to guess, enclosed in quotes) for a pair of negotiating hosts.
# Must be same on both; generate on one and copy to the other.
#10.0.0.1 10.12.12.1 : PSK
"jxRTnk3SlVu44l2U5U4umS1Sj1WSuVT4U14mlu3nURjR3mu24WmUWRVumRWkklj4unWj3UTkV"


192.168.33.27 192.168.33.26 "<insert password here>"
```

SSH Configuration

```
# This is ssh client system wide configuration file.  This file provides
# Defaults for users, and the values can be changed in per-user configuration
# Files or on the command line.

# Configuration data is parsed as follows:
#  1. command line options
#  2. user-specific file
#  3. system-wide file
# Any configuration value is only changed the first time it is set.
# Thus, host-specific definitions should be at the beginning of the
# configuration file, and defaults at the end.

# Site-wide defaults for various options

# Host *
#   ForwardAgent yes
#   ForwardX11 yes
#   RhostsAuthentication yes
#   RhostsRSAAuthentication yes
#   RSAAuthentication yes
#   TISAuthentication no
    PasswordAuthentication yes
#   FallBackToRsh yes
#   UseRsh no
#   BatchMode no
#   StrictHostKeyChecking no
#   IdentityFile ~/.ssh/identity
#   Port 22
#   Cipher idea
#   EscapeChar ~
```

IPChains Configuration

```
#!/bin/sh
# pmfirewall
# chkconfig: 2345 50 80
# description: Control script for pmfirewall package.
#

CONFIG_DIR=/usr/local/pmfirewall
# Source function library.
. /etc/rc.d/init.d/functions


## Read Configuration File
. $CONFIG_DIR/pmfirewall.conf

case "$1" in

#####START FIREWALL#####
start)
     echo -n "Starting PMFirewall:"
     ## Flush rule sets, start from scratch
     $IPChains -F input
     $IPChains -F output
     $IPChains -F forward

     ## Read firewall rules
     . $CONFIG_DIR/pmfirewall.rules.1
     . $CONFIG_DIR/pmfirewall.rules.local

     # Allow incoming and outgoing ICMP
     $IPChains -A input -p icmp -s $REMOTENET -d $OUTERNET -j ACCEPT
     $IPChains -A output -p icmp -s $OUTERNET -d $REMOTENET -j ACCEPT

     # These are open to sockets created by connections allowed by ipchains
     $IPChains -A input -p tcp -s $REMOTENET -d $OUTERNET 1023:65535 -j
ACCEPT
     $IPChains -A input -p udp -s $REMOTENET -d $OUTERNET 1023:65535 -j
ACCEPT

     ## Set default policy
     $IPChains -A output -j ACCEPT
     $IPChains -A input -j DENY -l
     echo "       Done!"
     echo ""
     echo "External: $OUTERIF $OUTERNET"
     echo "" ;;

#####STOP FIREWALL####
stop)
     echo ""
     echo -n "Shutting down PMFirewall:"
     $IPChains -F input
     $IPChains -F output
     $IPChains -F forward
     $IPChains -P forward DENY
     echo "       Done!"
     echo "" ;;

restart)
        $0 stop
        $0 start
        ;;

masqstart)
     echo ""
     echo "IP Masquerading was not enabled during the install process."
     echo ""
     echo "You must reinstall to use this option."
     echo ""
      ;;


masqstop)
     echo ""
     echo "IP Masquerading was not enabled during the install process."
     echo ""
     echo "You must reinstall to use this option."
     echo ""
      ;;


uninstall)
     $CONFIG_DIR/uninstall
     ;;


  *)

     echo ""
     echo "  USAGE:  pmfirewall [command] "
     echo ""
     echo "  COMMANDS:"
     echo "        start      Enables PMFirewall."
     echo "        stop       Disables PMFirewall."
     echo "        restart    Flushes and reloads the rules in PMFirewall."
     echo "        masqstart  Enables IP Masquerading only (no firewall)."
     echo "        masqstop   Disables IP Masquerading only (no firewall)."
     echo "        uninstall  Completely removes PMFirewall."
     echo "        help       Displays this list of options."
     echo ""
     exit 1 ;;

esac
exit 0

#!/bin/sh
# pmfirewall.conf - used by pmfirewall package
IPChains=/sbin/ipchains
ATBOOT=1
CONFIG_DIR=/usr/local/pmfirewall
OUTERIF=eth0
REMOTENET=0/0
OUTERIP=192.168.33.27
OUTERMASK=255.255.255.248
OUTERNET=$OUTERIP/$OUTERMASK

#!/bin/sh
# pmfirewall.rules.1 used by pmfirewall package
#
#### Start Firewall ####

## Allow loopback interface
$IPChains -A input -i lo -s 0/0 -d 0/0 -j ACCEPT
$IPChains -A output -i lo -s 0/0 -d 0/0 -j ACCEPT

# Allow packets with ack bit set, they are from an established connection.
$IPChains -A input ! -y -p tcp -s $REMOTENET -d $OUTERNET -j ACCEPT

# Block incoming IP Spoofing

# Turn on Source Address Verification

if [ -e /proc/sys/net/ipv4/conf/all/rp_filter ]
 then
```

```
  for f in /proc/sys/net/ipv4/conf/*/rp_filter
  do
   echo 1 > $f
  done
fi

#Turn on SYN COOKIES PROTECTION (Thanks Holger!)
if [ -e /proc/sys/net/ipv4/tcp_syncookies ]
 then
  echo 1 > /proc/sys/net/ipv4/tcp_syncookies
fi

# Now read pmfirewall.rules.local

#!/bin/sh
# pmfirewall.rules.local
# ver.PM1 (do not remove this line)

                   ### BEGIN SYSTEM DEFAULTS ###

# Block Nonroutable IP's from entering on the External Interface
$IPChains -A input -j DENY -s 10.0.0.0/8 -d $OUTERNET -i $OUTERIF
$IPChains -A input -j DENY -s 127.0.0.0/8 -d $OUTERNET -i $OUTERIF
$IPChains -A input -j DENY -s 172.16.0.0/12 -d $OUTERNET -i $OUTERIF
$IPChains -A input -j DENY -s 192.168.0.0/16 -d $OUTERNET -i $OUTERIF


# - Specific port blocks on the external interface -
# This section blocks off ports/services to the outside that have
# vulnerabilities. This will not affect the ability to use these services
# within your network.
#

# Back Orifice (logged)
$IPChains -A input -p tcp -s $REMOTENET -d $OUTERNET 31337 -j DENY -l
$IPChains -A input -p udp -s $REMOTENET -d $OUTERNET 31337 -j DENY -l

# NetBus (logged)
$IPChains -A input -p tcp -s $REMOTENET -d $OUTERNET 12345:12346 -j DENY -l
$IPChains -A input -p udp -s $REMOTENET -d $OUTERNET 12345:12346 -j DENY -l

# Trin00 (logged)
$IPChains -A input -p tcp -s $REMOTENET -d $OUTERNET 1524 -j DENY -l
$IPChains -A input -p tcp -s $REMOTENET -d $OUTERNET 27665 -j DENY -l
$IPChains -A input -p udp -s $REMOTENET -d $OUTERNET 27444 -j DENY -l
$IPChains -A input -p udp -s $REMOTENET -d $OUTERNET 31335 -j DENY -l

# Multicast
$IPChains -A input -s 224.0.0.0/8 -d $REMOTENET -j DENY
$IPChains -A input -s $REMOTENET -d 224.0.0.0/8 -j DENY


                       ### END SYSTEM DEFAULTS ###


                            #### EXAMPLES ###
```

```
### ALLOWED NETWORKS
# Add in any rules to specifically allow connections from hosts/nets that
# would otherwise be blocked.
#$IPChains -A input -s [trusted host/net] -d $OUTERNET <ports> -j ACCEPT

### BLOCKED NETWORKS
# Add in any rules to specifically block connections from hosts/nets that
# have been known to cause problems. These packets are logged.
#$IPChains -A input -s [banned host/net] -d $OUTERNET <ports> -j DENY -l

### BLOCK ICMP ATTACKS
#
#$IPChains -A input -b -i $OUTERIF -p icmp -s [host/net] -d $OUTERNET -j DENY
-l


                         #### END OF EXAMPLES ###



     ### AUTOMATICALLY GENERATED BY THE INSTALL SCRIPT ###

#UNRESTRICTED ACCESS
$IPChains -A input -s 192.168.33.16/27 -d $REMOTENET -j ACCEPT

#DHCP CLIENT BLOCK
$IPChains -A input -p udp -s $REMOTENET -d $REMOTENET 67:68 -i $OUTERIF -j
DENY

#FTP
$IPChains -A input -p tcp -s 192.168.22.13/32 -d $OUTERNET 20 -j ACCEPT
$IPChains -A input -p tcp -s 192.168.22.13/32 -d $OUTERNET 21 -j ACCEPT

#TELNET
$IPChains -A input -p tcp -s 192.168.22.13/32 -d $OUTERNET 23 -j ACCEPT

#SSH
$IPChains -A input -p tcp -s 192.168.22.13/32 -d $OUTERNET 22 -j ACCEPT

#IDENTD
$IPChains -A input -p tcp -s $REMOTENET -d $OUTERNET 113 -j REJECT
$IPChains -A input -p udp -s $REMOTENET -d $OUTERNET 113 -j REJECT

#NETBIOS
$IPChains -A input -p tcp -s $REMOTENET -d $REMOTENET 137:139 -i $OUTERIF -j
DENY
$IPChains -A input -p udp -s $REMOTENET -d $REMOTENET 137:139 -i $OUTERIF -j
DENY

#RIP
$IPChains -A input -p udp -s $REMOTENET -d $REMOTENET 520 -i $OUTERIF -j
REJECT

#NFS
$IPChains -A input -p tcp -s $REMOTENET -d $REMOTENET 2049 -i $OUTERIF -j
DENY -l
```

```
$IPChains -A input -p udp -s $REMOTENET -d $REMOTENET 2049 -i $OUTERIF -j
DENY -l

#XSERVER
$IPChains -A input -p tcp -s $REMOTENET -d $REMOTENET 5999:6003 -i $OUTERIF -
j DENY
$IPChains -A input -p udp -s $REMOTENET -d $REMOTENET 5999:6003 -i $OUTERIF -
j DENY

#!/bin/sh
#pmfirewall.rules.masq - used by pmfirewall package
#

## Masquerading

## Modules to help certain services

/sbin/depmod -a  >/dev/null 2>&1
/sbin/modprobe ip_masq_ftp  >/dev/null 2>&1
/sbin/modprobe ip_masq_raudio  >/dev/null 2>&1
/sbin/modprobe ip_masq_irc  >/dev/null 2>&1
/sbin/modprobe ip_masq_icq  >/dev/null 2>&1
/sbin/modprobe ip_masq_quake  >/dev/null 2>&1
/sbin/modprobe ip_masq_user   >/dev/null 2>&1
/sbin/modprobe ip_masq_vdolive  >/dev/null 2>&1

## Masquerading firewall timeouts: tcp conns 8hrs, tcp after fin pkt 60s, udp
10min
$IPChains -M -S 14400 60 600

## Set up kernel to enable IP masquerading
echo 1 > /proc/sys/net/ipv4/ip_forward

## Set up kernel to handle dynamic IP masquerading
echo 1 > /proc/sys/net/ipv4/ip_dynaddr

## Don't Masquerade internal-internal traffic
$IPChains -A forward -s $INTERNALNET -d $INTERNALNET -j ACCEPT

## Don't Masquerade external interface direct
$IPChains -A forward -s $OUTERNET -d $REMOTENET -j ACCEPT

## Masquerade all internal IP's going outside
$IPChains -A forward -s $INTERNALNET -d $REMOTENET -j MASQ

## Set Default rule on MASQ chain to Deny
$IPChains -P forward DENY

## Allow all connections from the network to the outside
$IPChains -A input -s $INTERNALNET -d $REMOTENET -j ACCEPT
$IPChains -A output -s $INTERNALNET -d $REMOTENET -j ACCEPT

# This section manipulates the Type Of Service (TOS) bits of the
# packet. For this to work, you must have CONFIG_IP_ROUTE_TOS enabled
# in your kernel

# Set telnet, www, smtp, pop3 and FTP for minimum delay
```

```
$IPChains -A output -p tcp -d 0/0 80 -t 0x01 0x10
$IPChains -A output -p tcp -d 0/0 22 -t 0x01 0x10
$IPChains -A output -p tcp -d 0/0 23 -t 0x01 0x10
$IPChains -A output -p tcp -d 0/0 21 -t 0x01 0x10
$IPChains -A output -p tcp -d 0/0 110 -t 0x01 0x10
$IPChains -A output -p tcp -d 0/0 25 -t 0x01 0x10

# Set ftp-data for maximum throughput
$IPChains -A output -p tcp -d 0/0 20 -t 0x01 0x08

# Allow outgoing ICMP
$IPChains -A output -p icmp -s $INTERNALNET -d $REMOTENET -j ACCEPT
```

Appendix D. Netscreen Configuration

The Netscreen is a firewall/VPN device. In the prototype configuration shown in figure A-3, the Netscreen sets up the IPSec tunnel between itself and the BlueCat (simulated spacecraft node). The IPSec tunnel provides authentication and confidentiality of all traffic between the Netscreen and BlueCat. The firewall capability of the Netscreen does not allow any other traffic other than the traffic sent through the tunnel to enter the trusted network. The Netscreen is configured mainly using the web GUI. The configuration of Netscreen-5XP used in the prototype is described below:

Connect the trusted interface of the Netscreen-5XP to the *Trusted Network* (192.168.33.16/29) and connect the untrusted interface to the *Space to Ground* network (192.168.33.0/29). Perform the initial configuration by running the nsqstart.exe (Quick Start) program on the Windows2000 machine (192.168.33.22). When the **Netscreen Quick Start (1/3) – Welcome** page appears, click next. When this program detects the Netscreen-5XP, the **Netscreen Quick Start (2/3)** – **Select Device** screen appears. Click the device on this screen and then click **Next**. The **Netscreen Quick Start (3/3) – Configuration** screen appears. Specify 192.168.33.19 as the **System IP Address** and select the **Network Address Translation Mode (NAT)** radio button. (Note that we want to configure this device to work in route mode. This will be done on the next screen). Click on the **Launch web browser for further configuration** check box. Click **Finish**. The **Netscreen Quick Start (4/4) – Configuration (NAT)** screen appears now. Specify 192.168.33.19, 255.255.255.248, and 1192.168.33.18 as the IP Address, Subnet, and Gateway respectively. Click the **Route Mode** radio button. Select the **Manually Assign** radio button and then assign 192.168.33.26 and 255.255.255.248 for IP Address and Subnet, respectively. Select **Launch web browser for further configuration** and click **Finish**. The login screen for the web interface now appears. The username and password for the initial login is **Netscreen**.

After logging on to the web GUI, all the configurations can be accomplished through the web GUI. For more detailed information on the Netscreen configuration, refer to the following documents:

- Netscreen Concepts & Examples ScreenOS Reference Guide
- Netscreen Web GUI Reference Guide
- Netscreen CLI Reference Guide

The Web GUI along with the specific values used to configure Netscreen-5XP for the prototype configuration is shown in Figure D-1 and described in the following text.

1. Activate the Netscreen configuration tool. It will open with a screen showing the status of any defined subnets and nodes. The screen has a series of buttons along the left-hand side that enable various features to be configured. A series of tabs at the top of the display allow various features to be configured. These tabs are "General", "Authen.", "DNS", "URL Filtering", "Route Table", "DHCP", and "Software Key". The startup screen is shown in figure D-1.

2. Clicking on the "Interface" button brings up a screen to set the parameters for a trusted interface.

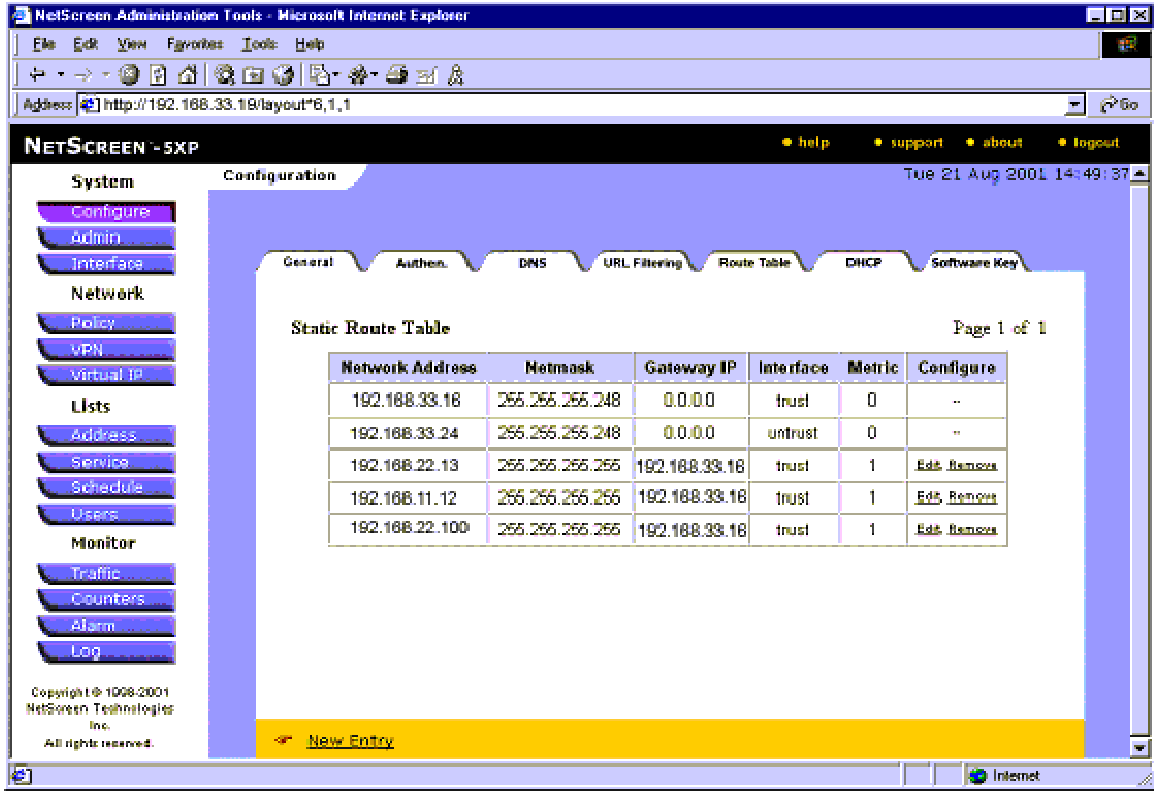
3. Clicking on the "Address" button brings up a screen to set the parameters for an interface. Two tabs at the top of the display allow switching from "Trusted" to "Untrusted" associations. Here names are associated with IP addresses.

4. Clicking on the "Policy" button brings up a screen showing the permitted outgoing tunnels. There are two tabs at the top of the display that allow switching form "Incoming" to "Outgoing" associations. Each summary line has an "Edit" button that allows changes to be made. There is a button to create a new policy at the bottom of the screen.

5. Clicking on the "Edit" button brings up a screen allowing the configuration of endpoints, guaranteed bandwidth, priority, etc.

6. Selecting "VPN" under the "Network" button grouping on the left brings up a screen to allow the configuration of phase 2 parameters for the IKE protocol for each tunnel.

7. Clicking on the "List Gateways" hot spot brings up a screen that allows the configuration of the phase 1 parameters for the gateway.



Static Route Table

Page 1 of 1

Network Address	Netmask	Gateway IP	Interface	Metric	Configure
192.168.33.16	255.255.255.248	0.0.0.0	trust	0	--
192.168.33.24	255.255.255.248	0.0.0.0	untrust	0	--
192.168.22.13	255.255.255.255	192.168.33.18	trust	1	Edit Remove
192.168.11.12	255.255.255.255	192.168.33.18	trust	1	Edit Remove
192.168.22.100	255.255.255.255	192.168.33.18	trust	1	Edit Remove

Figure D-1. Netscreen Configuration Startup Screen

Appendix E. Cisco PIX-515 Configuration

The PIX is configured to protect the trusted network from other CNE nodes and nodes on Internet. It is configured to allow any incoming connections from those nodes that have already established an IPSec tunnel with the PIX. The nodes on the trusted network are allowed to make connections to any machine on the untrusted network. The configuration also allows certain machine on the untrusted network to make connections to the nodes on the trusted network. For more detailed information on configuring the PIX firewall refer to the following two documents:

IPSec User Guide for the Cisco Secure PIX Firewall Version 6.0

Configuration Guide for the Cisco Secure PIX Firewall Version 6.0

The configuration file that is setup on the PIX-515 for the prototype configuration shown below:

```
PIX Version 6.0(1)
nameif ethernet0 untrusted security0
nameif ethernet1 inside security100
enable password 1CWzGbYlhg3X7uPx encrypted
passwd 2KFQnbNIdI.2KYOU encrypted
hostname pixie
domain-name gsfc.nasa.gov
fixup protocol ftp 21
fixup protocol http 80
fixup protocol h323 1720
fixup protocol rsh 514
fixup protocol smtp 25
fixup protocol sqlnet 1521
fixup protocol sip 5060
fixup protocol skinny 2000
names
access-list acl_out permit icmp any any echo-reply
access-list acl_out permit icmp any any unreachable
access-list acl_out permit tcp host 192.168.11.12 host 192.168.33.25 eq ftp
access-list acl_out permit icmp host 192.168.11.12 192.168.33.16
255.255.255.240
access-list acl_out permit icmp host 192.168.22.13 192.168.33.16
255.255.255.240
access-list 101 permit ip any host 192.168.11.12
pager lines 22
logging on
logging monitor debugging
logging buffered informational
interface ethernet0 100basetx
interface ethernet1 10baset
mtu untrusted 1500
mtu inside 1500
ip address untrusted 192.168.33.1 255.255.255.240
ip address inside 192.168.33.18 255.255.255.248
ip audit info action alarm
ip audit attack action alarm
ip local pool dealer 10.1.1.1-10.1.1.254
no failover
failover timeout 0:00:00
failover poll 15
failover ip address untrusted 0.0.0.0
failover ip address inside 0.0.0.0
pdm history enable
arp timeout 240
nat (inside) 0 192.168.33.16 255.255.255.248 0 0
nat (inside) 0 192.168.33.24 255.255.255.248 0 0
static (inside,untrusted) 192.168.33.24 192.168.33.24 netmask 255.255.255.248
0 0
static (inside,untrusted) 192.168.33.16 192.168.33.16 netmask 255.255.255.248
0 0
access-group acl_out in interface untrusted
route untrusted 0.0.0.0 0.0.0.0 192.168.33.2 1
route inside 192.168.33.24 255.255.255.248 192.168.33.19 1
timeout xlate 3:00:00
timeout conn 1:00:00 half-closed 0:10:00 udp 0:02:00 rpc 0:10:00 h323 0:05:00
sip 0:30:00 sip_media 0:02:00
timeout uauth 0:05:00 absolute
aaa-server TACACS+ protocol tacacs+
aaa-server RADIUS protocol radius
aaa-server anaro-auth protocol radius
aaa authentication include tcp/0 untrusted 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0
anaro-auth
no snmp-server location
no snmp-server contact
snmp-server community public
no snmp-server enable traps
floodguard enable
sysopt connection permit-ipsec
no sysopt route dnat
crypto ipsec transform-set tset1 esp-3des esp-sha-hmac
crypto ipsec transform-set tset2 esp-3des esp-md5-hmac
crypto ipsec transform-set strong-des esp-3des esp-sha-hmac
crypto dynamic-map cisco 4 set transform-set strong-des
crypto map partner-map 20 ipsec-isakmp dynamic cisco
crypto map partner-map interface untrusted
isakmp enable untrusted
isakmp key cisco1234 address 192.168.11.12 netmask 255.255.255.255
isakmp key cisco1234 address 192.168.33.191 netmask 255.255.255.255
isakmp key cisco1234 address 192.168.22.13 netmask 255.255.255.255
isakmp identity address
isakmp policy 10 authentication pre-share
isakmp policy 10 encryption 3des
isakmp policy 10 hash sha
isakmp policy 10 group 2
isakmp policy 10 lifetime 86400
isakmp policy 20 authentication pre-share
isakmp policy 20 encryption des
isakmp policy 20 hash sha
isakmp policy 20 group 1
isakmp policy 20 lifetime 86400
vpngroup omni address-pool dealer
vpngroup omni idle-time 1800
vpngroup omni password cisco1234
telnet 192.168.11.12 255.255.255.255 untrusted
telnet 192.168.33.182 255.255.255.255 inside
telnet timeout 30
```

```
ssh 192.168.11.12 255.255.255.255 untrusted
ssh timeout 15
terminal width 80
Cryptochecksum:cecc0489e30b9f7445f16665254a6dbc
: end
```

Appendix F. IPSec Configuration on Windows2000

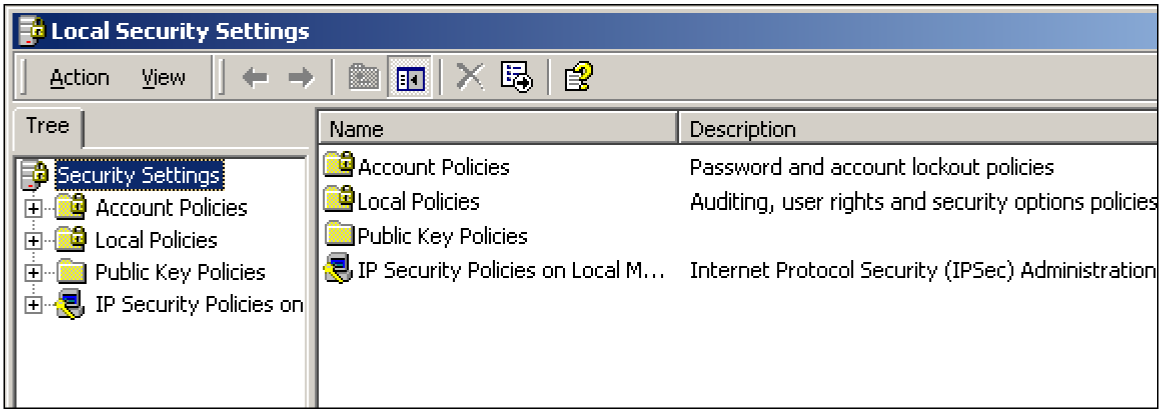
Windows2000 provides IPSec support. Since the IPSec support is native to Windows2000, no additional software needs to be purchased to provide IPSec support on Windows2000. One restriction on the IPSec tunnel set up is that both ends of the tunnel should have static IP address. For more details on how to configure IPSec Tunneling in Windows2000, go to: http://support.microsoft.com/support/kb/articles/Q252/7/35.ASP

Note that Cisco also provides a VPN client (3.0) that works on Windows2000 and other Windows operating systems. This client is mainly intended for remote dialup users where the remote machine does not necessarily have a static IP address. The configuration of this Cisco Client 3.0 is described in Appendix H. When this Cisco client is installed on Windows machine, all the IP traffic is sent through the tunnel. It is not possible to specify which traffic should use IPSec and which should not use IPSec. However, the native IPSec support provided by Windows2000 allows both IPSec protected traffic as well as traffic that is not protected by IPSec.

The advanced security option (128 bit) for Windows2000 needs to be installed, if the tunnel needs 3DES encryption. Otherwise, it did only DES encryption, although the configuration screen accepted 3DES. The advanced security option can be freely downloaded from Microsoft.

The pages below contain the opening window and the parameters that are specified for configuring the IPSec tunnel between Windows2000 (192.168.22.13) and the PIX firewall.

Select "Start" then "Run", enter "fsecpol.msc" and press the enter key. A screen as should appear as shown in Figure F-1.



Name	Description
Account Policies	Password and account lockout policies
Local Policies	Auditing, user rights and security options policies
Public Key Policies	
IP Security Policies on Local M...	Internet Protocol Security (IPSec) Administration

Figure F-1 Local Security Settings Window

1. Right Click "IP Security Policies on Local Machine", then select "Create IP Security Policy". The "IP Security Policy Wizard" window should appear.

2. Specify the name and description for the policy and click on the "Next" button.

3. In the window that opens, click to clear the "Activate the default response rule" check box and click the "Next" button.

4. In the widow that opens, ensure that the "Edit properties" check box is checked and click "Finish".

5. In the window with the name you specified for the security policy that opens, click to clear the "Use Add Wizard" check box on the "Rules" tab and click "Add" to create a new rule.

6. In the window that opens, select the "IP Filter List" tab and click "Add".

7. In the window that opens, enter a "Name" for the policy and click the "Use Add Wizard" check box to clear it. Then click the "Add..." button.

8. In the window that opens, in the "Source address" combo box select "My IP address" and in the "Destination address" combo box select "A specific IP Subnet". Then fill in the "IP Address" text box with the IP address for the trusted network with the mask in the "Subnet mask" text box. Click the "Mirrored. Also ..." check box to clear it.

9. Click on the "Protocol" tab and select "Any" for the "Select a protocol type:" combo box. Click on the "Description" tab if you want to enter a description of this filter. Then click the "OK" button and then the "Close" button in the window that opens.

10. In the window that opens, click "Add" to define the filter list from the PIX to Windows2000.

11. In the window that appears, enter a "Name" for the filter list and click the "Use Add Wizard" check box to clear it. Then click "Add".

12. In the "Addressing" tab of the "Filter Properties" window that opens, select "A specific IP Subnet" in the "Source Address combo box and fill in the "IP Address:" and "Subnet mask:" for the trusted subnet. Select "My IP Address" in the "Destination address:" combo box and click the "Mirrored. Also ..." check box to clear it.

13. Click the "Protocol" tab to select it and in the "Select a protocol type:" combo box select "Any". Click on the "Description" tab to enter a description of the filter. Then click "OK" and then "Close".

14. Select the Windows2000 to PIX rule radio button on the "IP Filter List" tab and then select the "Tunnel Setting" tab.

15. Select the "The tunnel endpoint is specified by this IP Address:" radio button and enter the outside IP address of the PIX firewall. Select the "Connection Type" tab.

16. Click on the "Local area network (LAN)" radio button and then select the "Filter Action" tab.

17. Click on the "Use Add Wizard" check box to clear it and click the "Add..." button.

18. In the window that appears, select the "Negotiate security" radio button and clear all the check boxes at the bottom of the window. Click the "Add..." button.

19. In the window that appears, select the "Custom (for expert users)" radio button and then click the "Settings..." button.

20. In the window that appears, select the "Data integrity and encryption (ESP):" check box, select "SHA1" for the "Integrity algorithm:" and select "3DES" for the "Encryption algorithm:". Then click the "OK" button.

21. In the window that appears, select the "General" tab, enter a "Name" for the filter action and click the "OK" button.

22. In the "New Rule Properties" window that appears, select the radio button for the filter action that was just created and then select the "Authentication Methods" tab.

23. In the window that opens, click the "Edit..." button.

24. In the window that appears, select the "Use this string to protect the key exchange (preshared key):" radio button and enter the key to be used between the PIX and Windows2000. Click the "OK" button and then the "Close" button in the new window.

25. In the window that appears, click "Add..." to create the IP filter list from the PIX to Windows2000.

26. In the "New Rule Properties" window that appears, select the PIX to Windows2000 IP filter list that was created earlier and select the "Tunnel Setting" tab.

27. Select "The tunnel endpoint is specified by this IP Address:" radio button and enter the IP address of the Windows2000 machine.

28. Select the "Connection Type" tab and click the "Local area network (LAN)" radio button.

29. Select the "Filter Action" tab and select the filter action that was created earlier.

30. Select the "Authentication Methods" tab and click "Edit...".

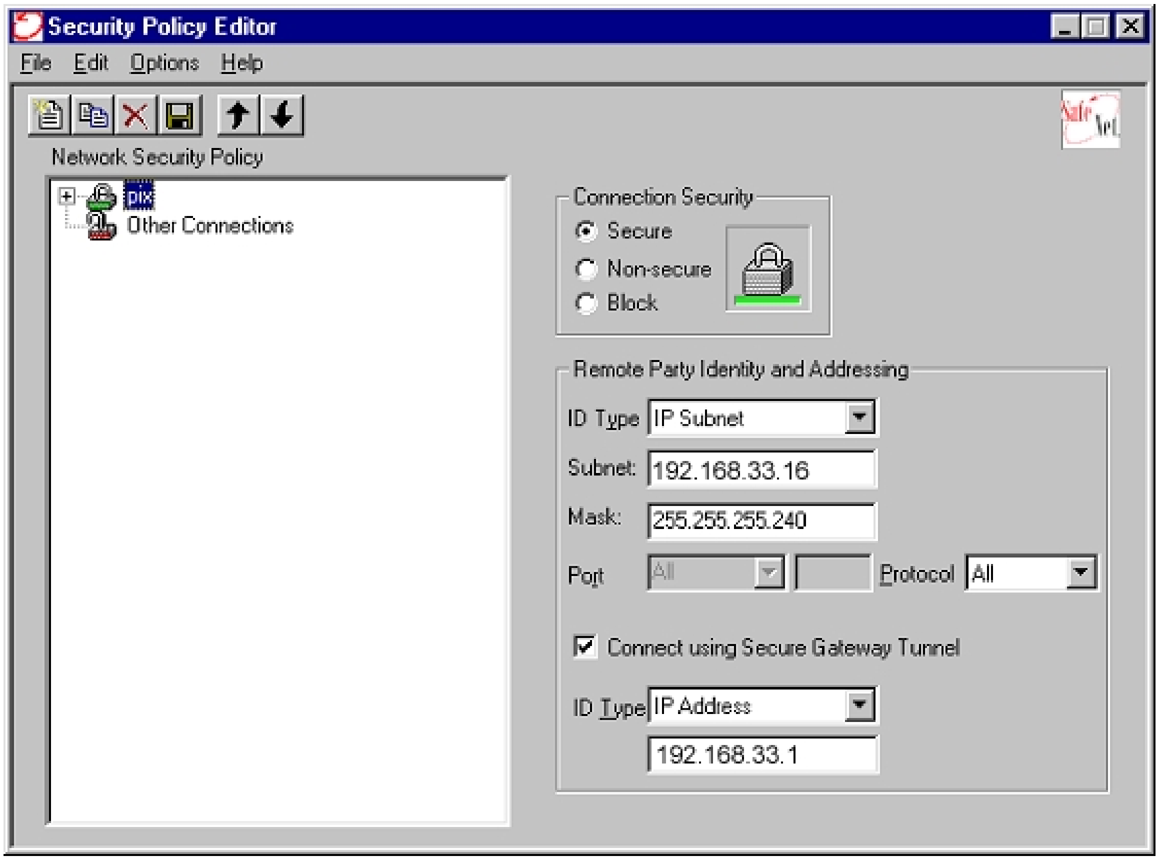
31. Configure this method the same as for the previous authentication method. They must be the same to enable communications.

The configuration is now complete.

Appendix G. Cisco Secure VPN Client 1.1 IPSec Configuration on Windows95

IPSec tunnel is established between the PIX firewall and the Windows95 machine (192.168.11.12) in Greentech IV is established using the Cisco Secure VPN Client 1.1. The procedure to configure this client is described below:

Click Start>Programs>Cisco Secure VPN Client>Security Policy Editor. This brings up a GUI. Enter the parameters and selections as shown in the display below:



Connection Security
● Secure
○ Non-secure
○ Block

Remote Party Identity and Addressing
ID Type: IP Subnet
Subnet: 192.168.33.16
Mask: 255.255.255.240
Port: All
Protocol: All
☑ Connect using Secure Gateway Tunnel
ID Type: IP Address
192.168.33.1

Figure G-1 Cisco Secure VPN Client Screen

1. In the window that appears, select the "Secure" radio button, select "IP Subnet" in the "ID Type" combo box, enter the IP address for the trusted subnet, enter the mask for the trusted subnet in "Mask", select "All" in the "Protocol" combo box, select the "Connect using Secure Gateway Tunnel" check box, select "IP Address" in the "ID Type" combo box, enter the untrusted network address in the text box, and then expand the "pix" item in the left pane by clicking the + next to it.

2. In the left pane of the window that opens, click "My Identity" and complete the right pane by selecting "None" for the "Select Certificate" combo box, selecting "IP Address" in the "ID Type" combo box, and "Any" in the "Name" combo box. Then click the "Pre-Shared Key" button.
3. In the window that appears, click the "Enter Key" button, enter the value of the pre-shared secret key used between the PIX and VPN client, and click on the "OK" button.
4. In the left pane of the window that opens, click the "Security Policy" item and select the "Main Mode" radio button, select the "Enable Perfect Forward Secrecy (PFS)" check box, and select "Diffie-Hellman Group 2" in the "PFS Key Group" combo box.
5. Expand the "Security Policy" item by clicking the + on its left, then expand "Authentication (Phase 1)" by clicking the + on its left and select the "Proposal 1" subitem under the "Authentication (Phase 1)" item. In the right pane select "Pre-Shared Key" in the "Authentication Method" combo box, select "Triple DES" in the Encrypt Alg" combo box, select "SHA-1" in the "Hash Alg" combo box, select "Unspecified" in the "SA Life" combo box and "Diffie-Hellman Group 2" in the "Key Group" combo box.
6. Expand the "Key Exchange (Phase 2)" item in the left pane by clicking the + on its left and select the "Proposal 1" subitem. In the right pane select "Unspecified" in the "SA Life" combo box, select the "Encapsulation Protocol (ESP)" check box, select "Triple DES" in the "Encrypt Alg" combo box, select "SHA-1" in the "Hash Alg" combo box, and "Tunnel" in the "Encapsulation" combo box.
7. Click the save button to save the changes and exit the configuration GUI.

After this configuration is complete, any connections from this Window95 machine to any node on the 192.168.33.16/28 segment establish an IPSec tunnel between this machine and the PIX firewall. A log of the IPSec key exchange and connection establishment can be displayed by activating the log viewer. Right clicking the small VPN client icon that appears on the task bar activates the log viewer.

Appendix H. Configuring Cisco VPN 3000 Client or a Cisco VPN Client Version 3.0/3.1

IPSec tunnel is established between the PIX firewall and the Windows95 machine (192.168.11.12) in Greentech IV is established using the Cisco Secure VPN Client 3.0. The procedure to configure this client is described below. The configuration these two clients are same. The GUI and the parameters that are shown on following pages apply to both of them.

Bring up the VPN client by **Start>Programs>Cisco Systems VPN Client>VPN Dialer**. The following window appears on the screen.

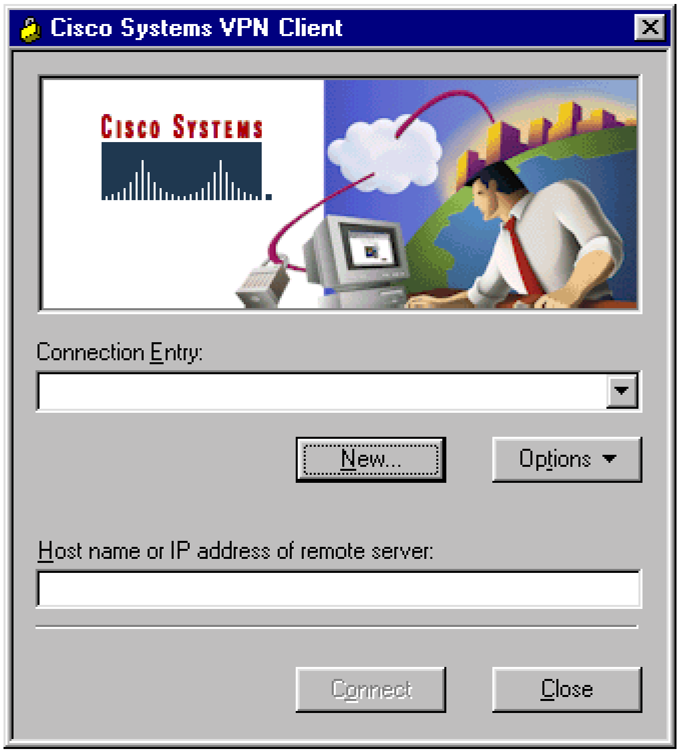


Figure H-1. The Cisco VPN Client Opening Screen

1. Click "New" to configure a new connection.
2. In the window that appears, enter a name for the connection entry in "Name of the new connection entry:" and an optional description in the text box below. Click the "Next" button.
3. In the window that appears, enter the IP address of the PIX firewall in "Host name or IP address of the server:" and click the "Next" button.
4. In the window that appears, enter a "Name:", "Password:", and "Confirm Password:" for the "Group Access Information" after enabling the associated radio button. Click the "Next" button.

5. A window displays saying the connection was successfully created. Click on the "Finish" button.

6. The initial window reappears with the "Connect" button enabled. Click this button to establish a VPN connection. After establishing the VPN, the VPN can be disconnected by right clicking the lock icon that appears on the task bar and then selecting the disconnect option.

Appendix I. IPSEC AH/ESP Procedure for Windows 2000

The following procedure is from the October 2000 issue of Windows 2000 Magazine with a few additions and minor corrections.

This process is lengthy and detailed. Fortunately, you can take advantage of Win2K's many IPSEC wizards, so make sure the Use Add Wizard option is selected on all the windows in which it appears. You need to create this Lockdown policy on both machines so that you can test whether the policy works. Here's the step-by-step procedure.

1. On system A, start Microsoft Management Console (MMC) and load the IP Security Policy Management snap-in for the local computer. To load the MMC, go to "Run" and enter “MMC”. Select the “Console” menu item and then the "Add/Remove Snap-in..." menu item. Click on "Add..." and then select the "IP Security Policy Management" snap-in in the pop-up window. Click "Add". Make sure that the “Local computer” radio button is selected in the “Select Computer” window and click “Finish”. In the “Add Standalone Snap-in” window click “Close”. In the “Add/Remove Snap-in” window click “OK”.
2. Right-click “IP Security Policies on Local Machine”, and click “Create IP Security Policy.”
3. On the welcome screen of the IP Security Policy Wizard, click “Next”.
4. Type “Lockdown” as the name of the policy, and click Next.
5. Clear the Activate the default response rule check box, and click “Next”.
6. Confirm that the “Edit Properties” check box is selected, and click “Finish”.
7. On the Lockdown Properties window, confirm that the <Default Response> rule check box is cleared. Confirm that the “Use Add Wizard” check box (in the lower right corner of the screen) is selected, and click “Add” to start the Security Rule Wizard.
8. Click “Next” to advance to the next screen of the Security Rule Wizard.
9. Select the “This rule does not specify a tunnel” check box, and click “Next”.
10. Select the Local area network (LAN) check box, and click Next.
11. On the Edit Authentication Method Properties dialog box select “Use this string to protect the key exchange (preshared key”), enter “lockdown” as the string, and click “Next”.
12. On the IP Filter List dialog box, click “Add”.
13. Name the filter Orange, confirm that “Use Add Wizard” is selected, and click “Add” to start the IP Filter Wizard.
14. Click “Next”.
15. Leave “My IP Address” as the Source address, and click “Next”.
16. Choose “A specific IP Address” from the Destination address drop-down list, enter the other computer's IP address, and click “Next”.
17. On the IP Protocol Type window, leave Any as the protocol type, and click “Next”.

18. On the Completing the IP Filter Wizard window, confirm that the “Edit Properties” check box is cleared, and click “Finish”.
19. Click “Close” to return to the IP Filter List window.
20. Select the radio button next to the Orange filter, and click “Next”.
21. On the Filter Action window, confirm that “Use Add Wizard” is selected, and click “Add”.
22. Click “Next”.
23. Name the Filter Action Orange Juice, and click “Next”.
24. On the Filter Action General Options window, select “Negotiate security, and click Next.
25. Select “Do not communicate with computers that do not support IPSEC”, and click “Next”.
26. Select “High (Encapsulated Secure Payload)”, and click “Next”. Here is where “AH” can be selected.
27. On the Completing the IP Security Filter Action Wizard window, confirm that “Edit Properties” is cleared and click “Finish”.
28. Select the Orange Juice radio button, and click “Next”.
29. On the Completing the New Rule Wizard window, confirm that “Edit Properties” is cleared, and click “Finish”.
30. On the Lockdown Properties window, confirm that the new Orange filter is selected. Click “Close” to complete the policy.
31. Repeat the procedure on system B. Be sure that you enter system A's IP address in step 15.
32. On system B, start Microsoft Management Console (MMC) and load the IP Security Policy Management snap-in for the local computer. To load the MMC, go to "Run" and enter “MMC”. Select the “Console” menu item and then the "Add/Remove Snap-in ..." menu item. Click on "Add..." and then select the "IP Security Policy Management" snap-in in the pop-up window. Click "Add".
33. Right-click “IP Security Policies on Local Machine”, and click “Create IP Security Policy.”
34. On the welcome screen of the IP Security Policy Wizard, click “Next”.
35. Type “Lockdown” as the name of the policy, and click Next.
36. Clear the Activate the default response rule check box, and click “Next”.
37. Confirm that the “Edit Properties” check box is selected, and click “Finish”.
38. On the Lockdown Properties window, confirm that the <Default Response> rule check box is cleared. Confirm that the “Use Add Wizard” check box (in the lower right corner of the screen) is selected, and click “Add” to start the Security Rule Wizard.
39. Click “Next” to advance to the next screen of the Security Rule Wizard.
40. Select the “This rule does not specify a tunnel” check box, and click “Next”.
41. Select the Local area network (LAN) check box, and click Next.

42. On the Edit Authentication Method Properties dialog box select “Use this string to protect the key exchange (preshared key”), enter “lockdown” as the string, and click “Next”.

43. On the IP Filter List dialog box, click “Add”.

44. Name the filter Orange, confirm that “Use Add Wizard” is selected, and click “Add” to start the IP Filter Wizard.

45. Click “Next”.

46. Choose “A specific IP Address” from the Source address drop-down list, enter the other computer's IP address, and click “Next”.

47. Leave “My IP Address” as the Destination address, and click “Next”.

48. On the IP Protocol Type window, leave Any as the protocol type, and click “Next”.

49. On the Completing the IP Filter Wizard window, confirm that the “Edit Properties” check box is cleared, and click “Finish”.

50. Click “Close” to return to the IP Filter List window.

51. Select the radio button next to the Orange filter, and click “Next”.

52. On the Filter Action window, confirm that “Use Add Wizard” is selected, and click “Add”.

53. Click “Next”.

54. Name the Filter Action Orange Juice, and click “Next”.

55. On the Filter Action General Options window, select “Negotiate security, and click Next.

56. Select “Do not communicate with computers that do not support IPSEC”, and click “Next”.

57. Select “High (Encapsulated Secure Payload)”, and click “Next”. Here is where “AH” can be selected.

58. On the Completing the IP Security Filter Action Wizard window, confirm that “Edit Properties” is cleared and click “Finish”.

59. Select the Orange Juice radio button, and click “Next”.

60. On the Completing the New Rule Wizard window, confirm that “Edit Properties” is cleared, and click “Finish”.

61. On the Lockdown Properties window, confirm that the new Orange filter is selected. Click “Close” to complete the policy.

62. Click on “IP Security Policies on Local Machine” in the left-hand window of the MMC and select the policy in the right-hand MMC window. Click on the “Action” item in the menu bar and select “Assign”.

63. Open the “Services” control panel and restart the “IP Security Agent” to make sure that the new policy is activated.

64. Go to workstation A and perform steps 62 and 63 there.

65. After testing, if unsecured communications are wanted, start the MMC as in step 1 and click on “IP Security Policies on Local Machine” in the left pane.

66. Right-click “Client (Respond Only)” in the right pane and select “Assign” from the drop-down menu. This action will assign the policy to allow normal communications.

Appendix J. Abbreviations and Acronyms

3DES	Triple DES
AC	Access Control
ADM	Administrative
AES	Advanced Encryption Standard
AH	Authentication Header
AMD	Advanced Micro Devices
ARTI	Advance Range Technology Initiative
Au	Authentication / Non-Repudiation
Av	Availability
BRT	Business and Restricted Technology
BSD	Berkeley Standard Distribution
C	Confidentiality
CBC	Cipher Block Chaining
CC	Control Center
CLI	Command Line Interface
CMN	Closed Mission Network
CMT	Cryptographic Module Testing
CNE	Center Network Environment
COTS	Commercial Off the Shelf
CPU	Central Processor Unit
CSC	Computer Sciences Corporation
DES	Data Encryption Standard
DI	Data Integrity
DNS	Domain Name Service
DSA	Digital Signature Algorithm
EEPROM	Electronically Erasable Programmable Read-Only Memory
ESP	Encapsulation Security Payload
FIPS	Federal Information Processing Standard

FOT	Flight Operations Team
FPGA	Field Programmable Gate Array
FTP	File Transfer Protocol
GKMP	Group Key Management Protocol
GSFC	Goddard Space Flight Center
GUI	Graphical User Interface
HST	Hubble Space Telescope
IACR	International Association for Cryptologic Research
ICMP	Internet Control Message Protocol
IDS	Intrusion Detection System
IETF	Internet Engineering Task Force
IKE	Internet Key Exchange
IONet	IP Operational Network
IP	Internet Protocol
IPSec	IP Security
ISAKMP	Internet Secure Association Key Management Protocol
ISO	International Standards Organization
IT	Information Technology
KB	Kilobyte
KBps	Kilobyte per second
LAN	Local Area Network
LST	Light Speed Test
MB	Megabyte
MD5	Message Digest 5
MSN	Mission
MTU	Maximum Transfer Unit
NASA	National Aeronautics and Space Administration
NAT	Network Address Translation
NCC	New Control Center
NIAP	National Information Assurance Partnership

NIC	Network Interface Card
NIST	National Institute of Standards & Technology
NPG	NASA Policy Guideline
NSA	National Security Agency
OMN	Open Mission Network
OMNI	Operating Missions as Nodes on the Internet
ON	Open Network
OSI	Open Systems Interconnection
PC	Personal Computer
PCI	Personal Computer Interface
PDM	PIX Device Manager
PI	Principal Investigator
PKI	Public Key Infrastructure
PUB	Public Access
RADIUS	Remote Access Dial-in User Service
RF	Radio Frequency
RFC	Request for Comments
RSA	Rivest-Shamir-Adleman Corporation
SA	Security Association
SANS	System, Administration, Networking, and Security
SCP	Secure Copy
SER	Scientific, Engineering, and Research
SFTP	Secure FTP
SHA	Secure Hash Algorithm
SKEME	Secure Key Exchange Mechanism
SKIP	Simple Key Management for Internet Protocol
SOMO	Space Operations Management Office
SP	Service Pack
SSH	Secure Shell
SSL	Secure Socket Layer

TCP	Transport Control Protocol
TDRS	Tracking and Data Relay Satellite
TFI	Traffic Flow Integrity
TFTP	Trivial File Transfer Protocol
TLS	Transport Layer Security
UDP	Unreliable Datagram Protocol
VPN	Virtual Private Network
WAN	Wide Area Network
WFF	Wallops Flight Facility

Appendix K. References

Brenton, Chris, Stephen Northcutt, Gary Kessler, Hal Pomeranz. VPN/Firewall Training Guide. SANS Institute.

Doraswamy, Naganand, and Dan Harkins. IPSEC: The New Security Standard for the Internet, Intranets, and Virtual Private Networks. Prentice Hall PTR, 1999.

Kaufman, Charlie, Radia Perlman, Mike Speciner. Network Security: Private Communication in a Public World. Prentice Hall, 1995.

Kilian, Joe. Advances in Cryptology. IACR (International Association Cryptologic Research) Crypto 2001 Proceedings.

Stallings, William. Network Security Essentials: Applications and Standards. Prentice Hall, 2000.

HyperLink References for IPSec are:

http://www.ietf.org/html.charters/ipsec-charter.html

http://www.tisc2001.com/newsletters/39.html

http://www.crypto-central.com/html/main.html

http://munitions.polkaroo.net/dolphin.cgi?action=render&category=01

HyperLink References for Linux and FreeS/WAN are:

http://www.linuxdoc.org/

http://www.freeswan.org/

RFCs 2401, 2402, 2406 and 2408 were referenced. HyperLink Reference for RFCs is:

http://www.rfc-editor.org/rfc.html